

Fig. 1

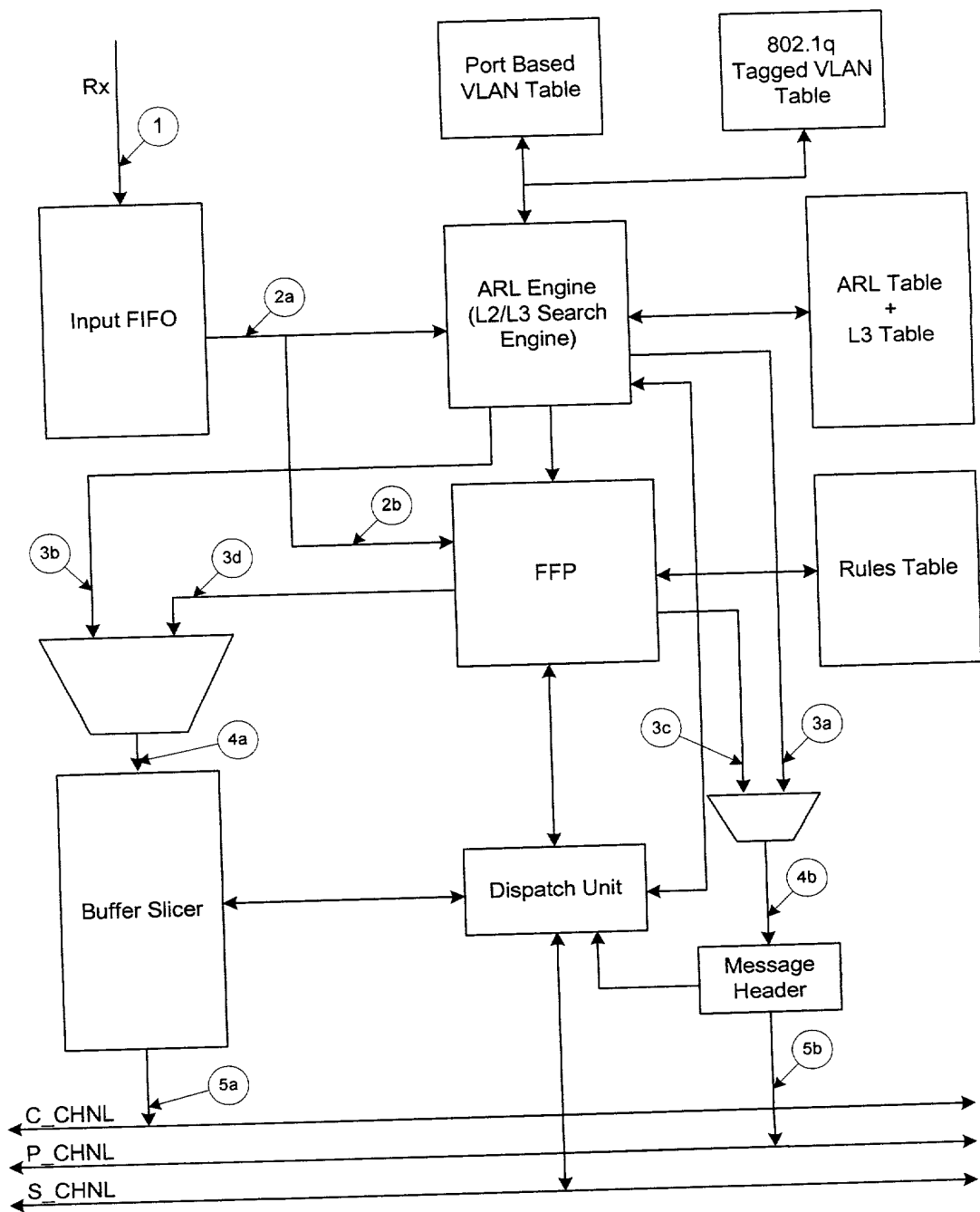


Fig. 2

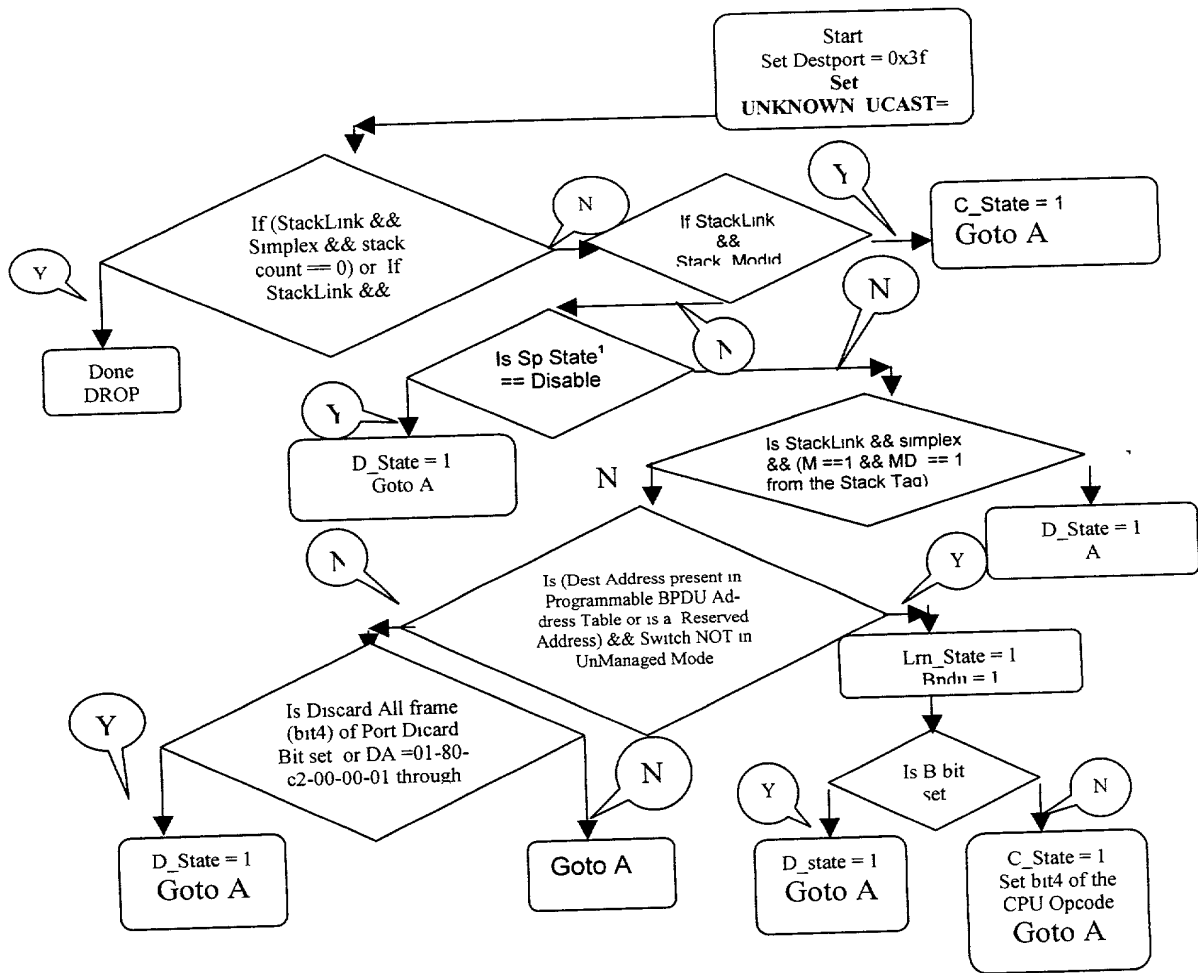


Fig. 3

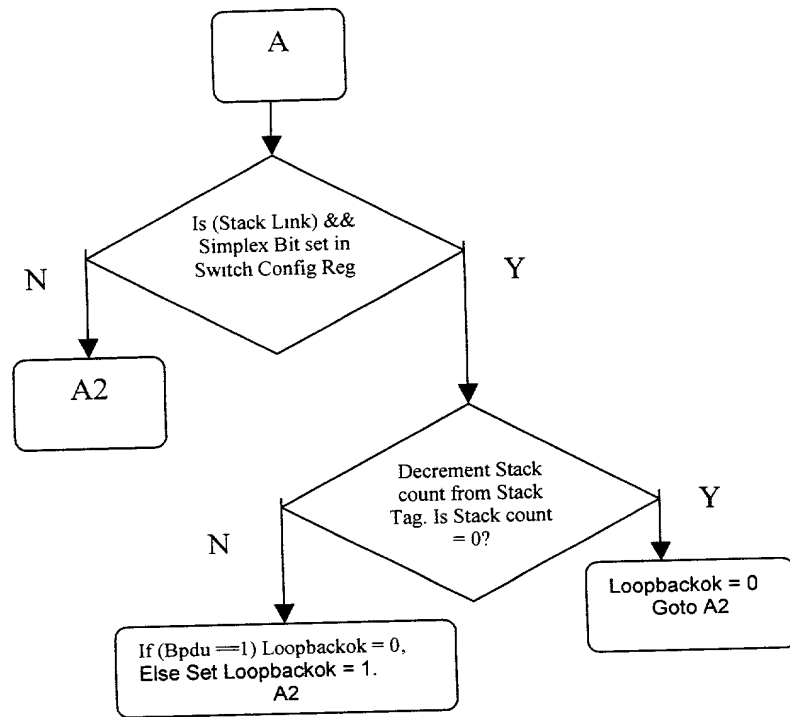


Fig. 4

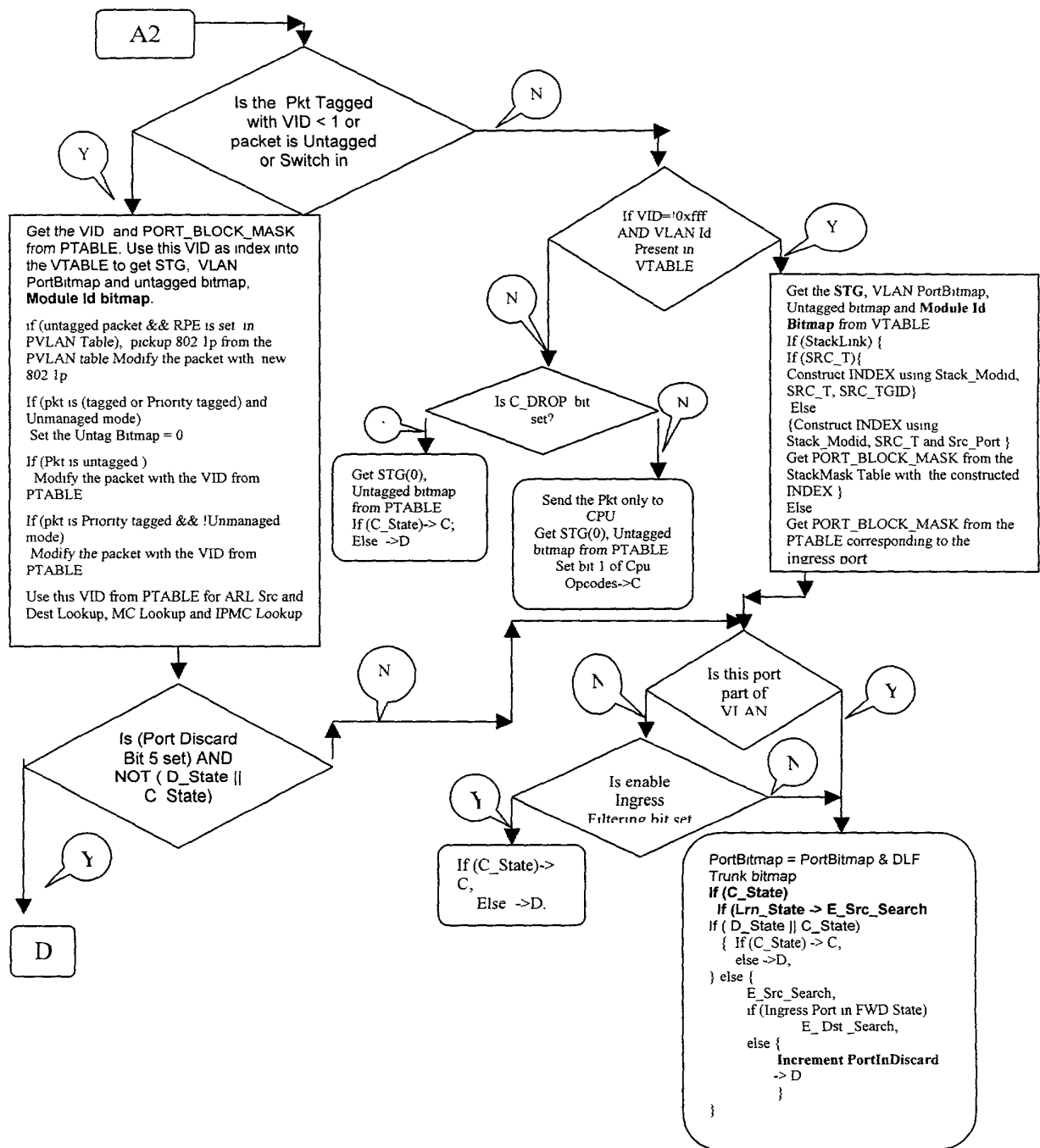


Fig. 5

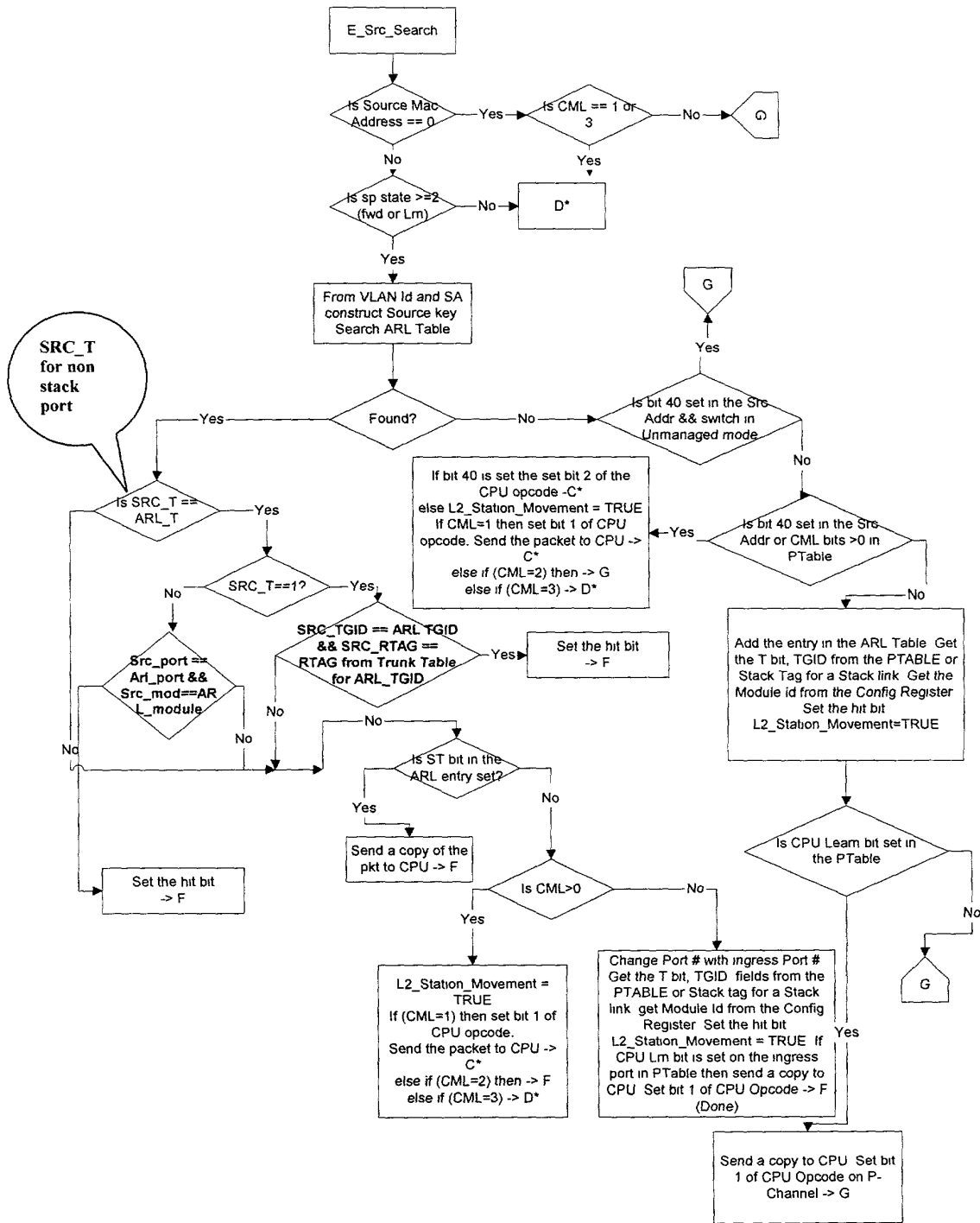


Fig. 6

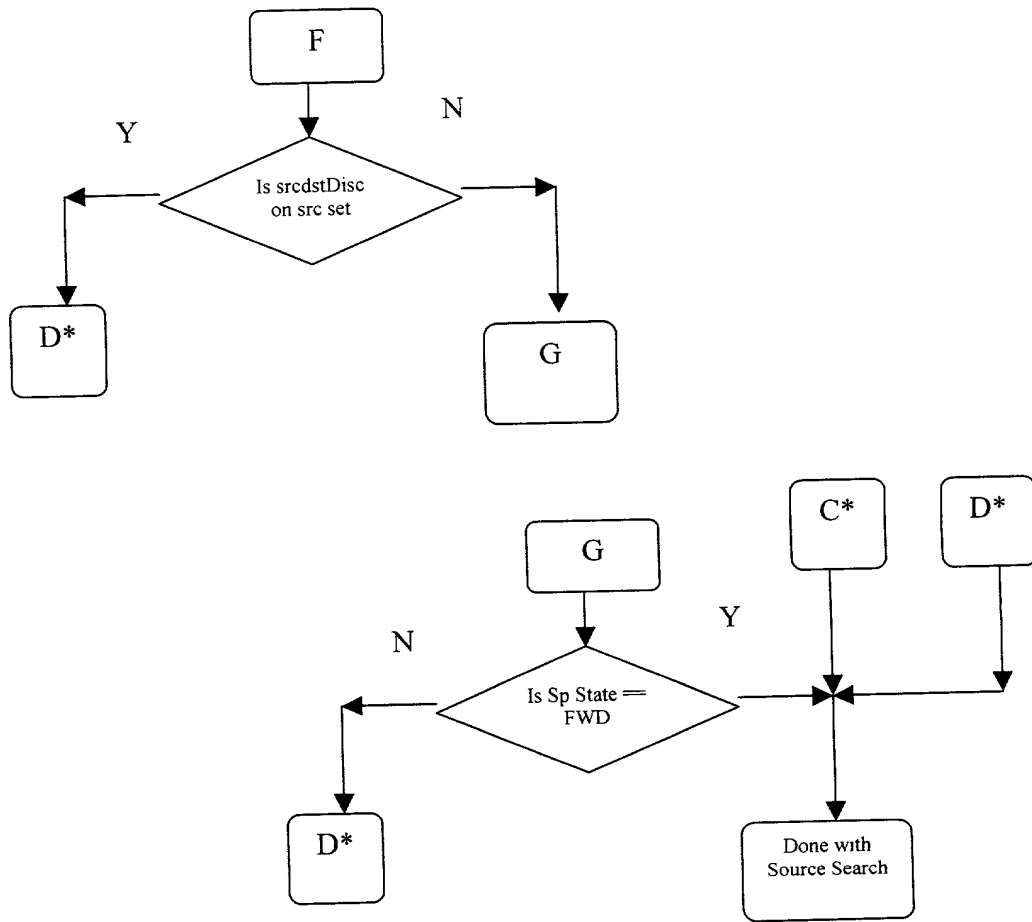


Fig. 7

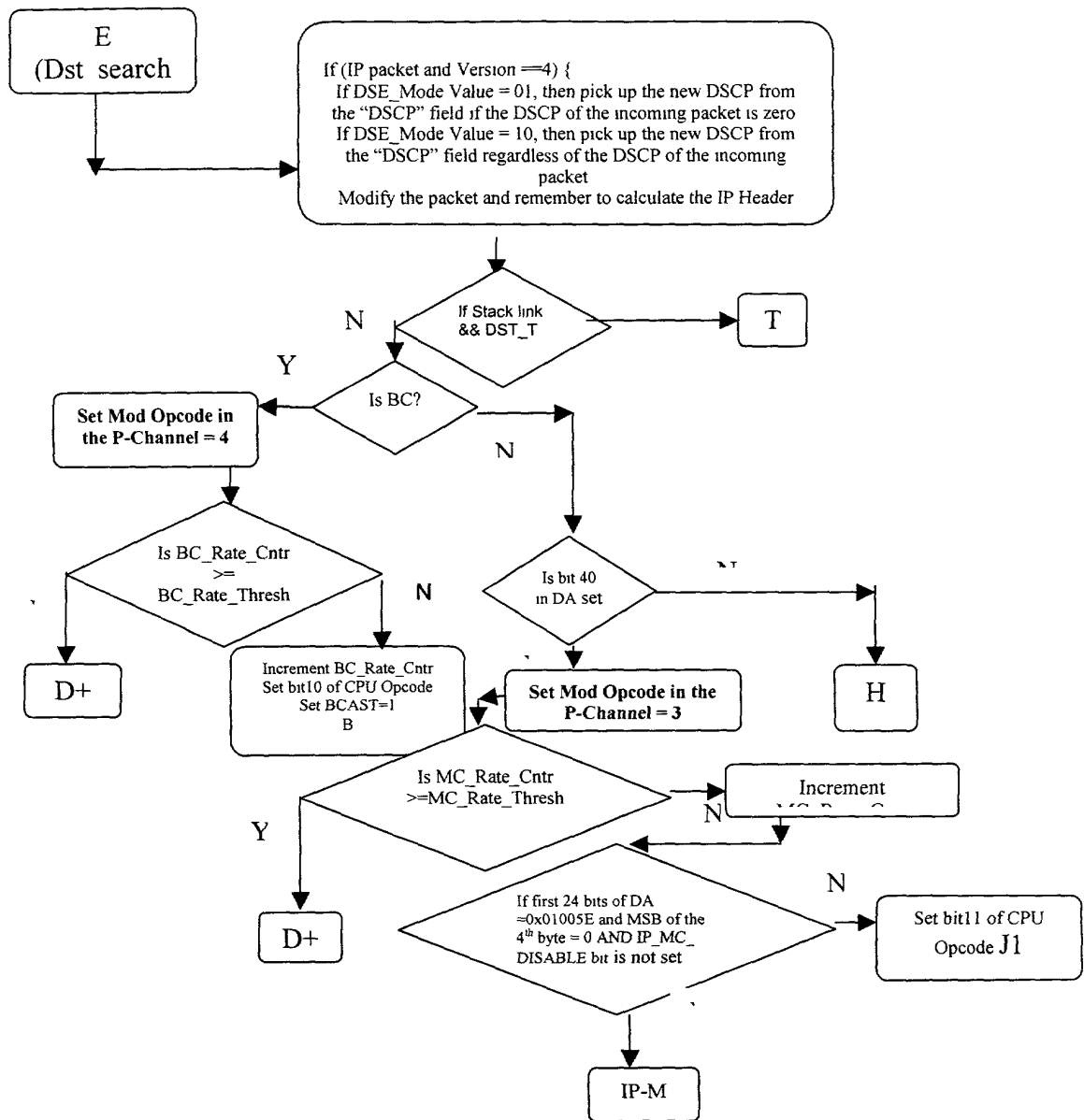


Fig. 8

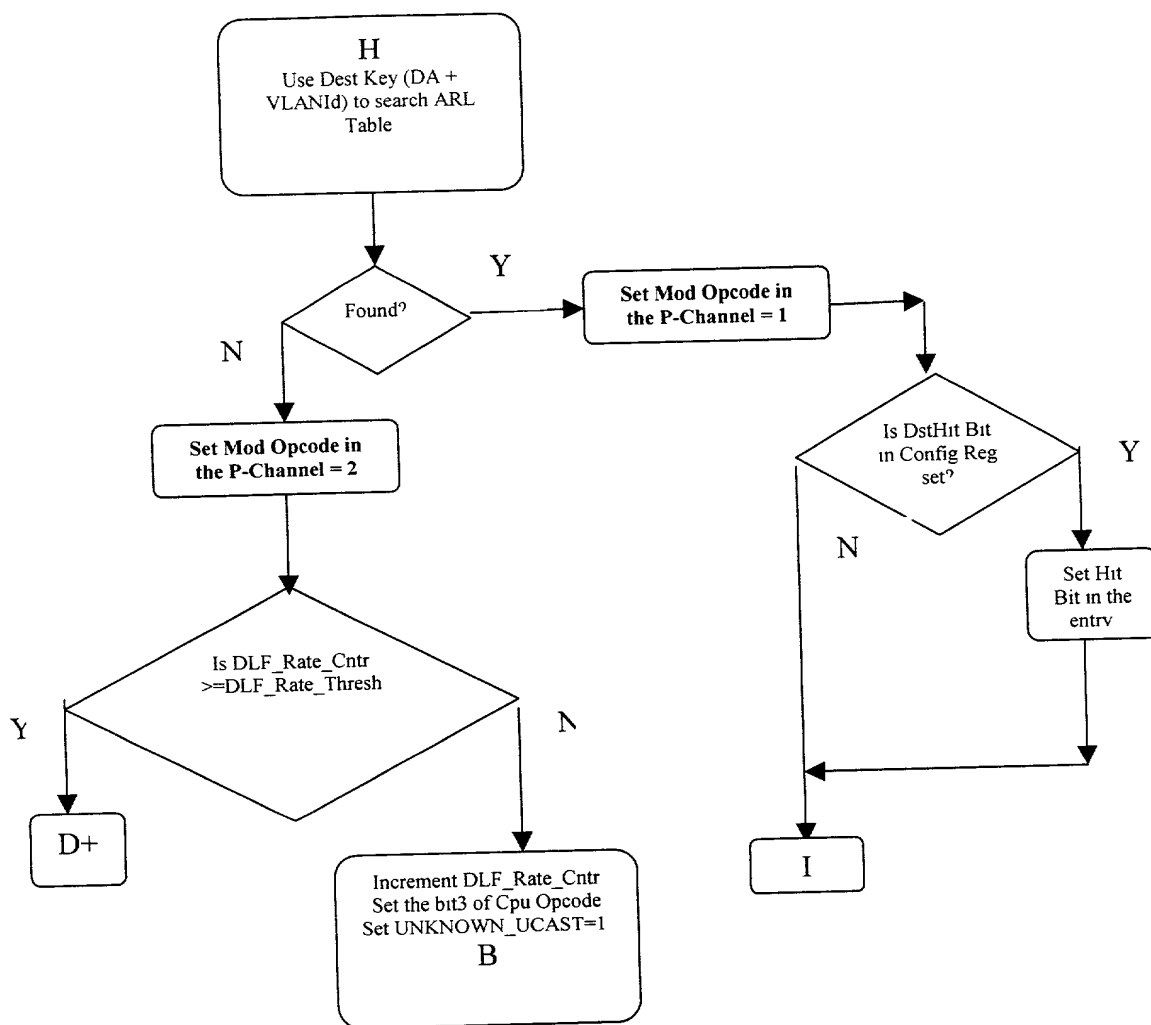


Fig. 9

```

graph TD
    I[I] --> L3_ArL{Is L3 Bit in ARL Entry set}
    L3_ArL -- N --> C_Bit{Is C Bit}
    L3_ArL -- Y --> StackLink{Is StackLink}
    C_Bit -- Y --> CPU[Send a copy of the Pkt to CPU]
    C_Bit -- N --> SrcDstDiscard{Is SrcDstDiscard Bit1 set?}
    StackLink -- Y --> Mirrored1[Remember to send the packet to Mirrored]
    StackLink -- N --> L3_Reg{Is L3 bit in Mirroring Reg set}
    SrcDstDiscard -- Y --> Dp1[Go to D+]
    SrcDstDiscard -- N --> StackLink2{Is Stack Link}
    L3_Reg -- Y --> Mirrored2[Remember to send the packet to Mirrored]
    L3_Reg -- N --> L3[L3]
    StackLink2 -- Y --> StackLink3[Stack-Link]
    StackLink2 -- N --> T_Bit{Is T Bit set}
    T_Bit -- Y --> T[T]
    T_Bit -- N --> Process1[Pick up Egress Port and Module Id in ARL entry  
If (module id != my_modid) {  
  Use IPIC port to construct Port Bitmap  
Else  
  Use Egress port to construct PortBitmap }  
Use Module Id to construct Module Id Bitmap]
    T --> Process2[If DST_T is set in Stack Tag then get TGID & RTAG from Stack Tag, else get TGID from ARL entry. Index into the Trunk Group Table with TGID and get RTAG. From RTAG get index into Trunk Group identified by TGID. Construct PortBitmap that includes this port. Construct the Module Id map including the Destination module.]
    Process1 --> J
    Process2 --> J
    J[PortBitmap = PortBitmap && (VLAN Bitmap for the VLAN)  
Exclude Source Module from the Module Id map. Exclude Source port from PortBitmap. If Source Port is a Trunk Port or Stack Link (refer to note2) then PortBitmap = PortBitmap & ~(Trunk Group Bitmap) for the Trunk identified by the SRC_TGID respectively. If !PortBitmap then go to D+] --> Dp2[Go to D+]
  
```

When the module id from the ARL is not my_modid, Include the IPIC in the Port Bitmap and not the egress port from the ARL Table

PortBitmap = PortBitmap && (VLAN Bitmap for the VLAN)
Exclude Source Module from the Module Id map. Exclude Source port from PortBitmap. If Source Port is a Trunk Port or Stack Link (refer to note2) then PortBitmap = PortBitmap & ~(Trunk Group Bitmap) for the Trunk identified by the SRC_TGID respectively. If !PortBitmap then go to D+

Fig. 10

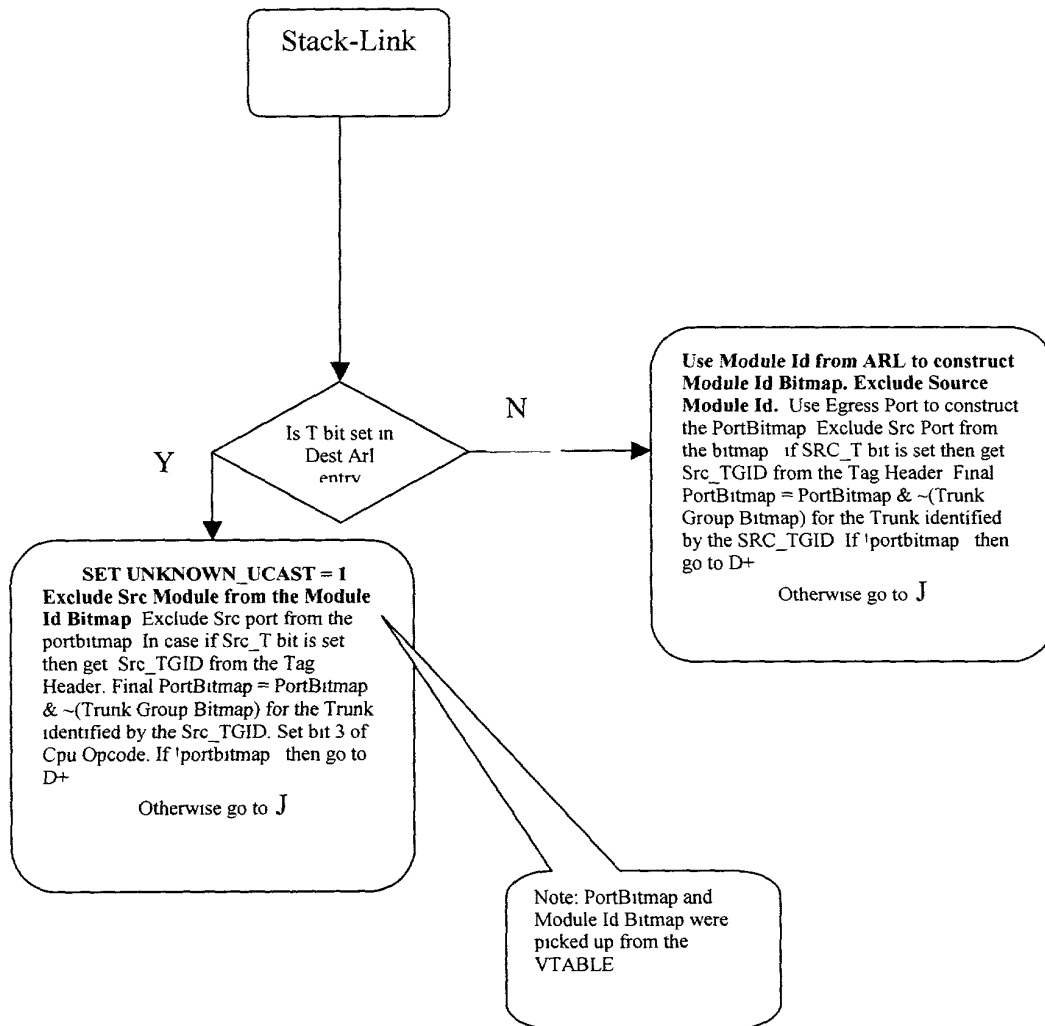


Fig. 11

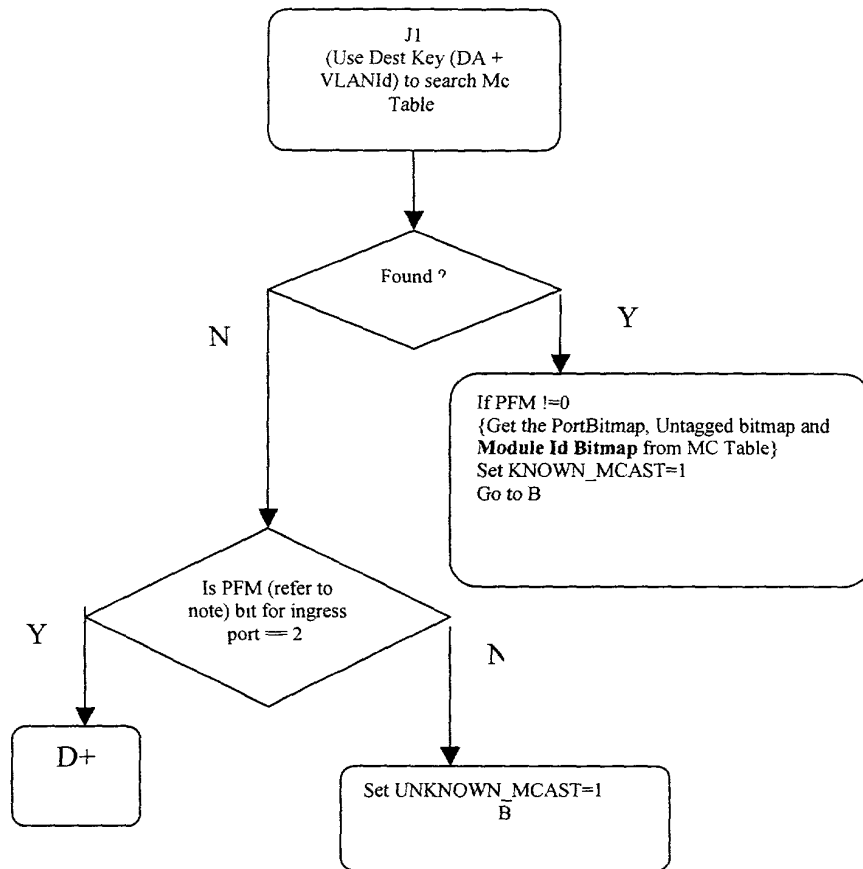


Fig. 12

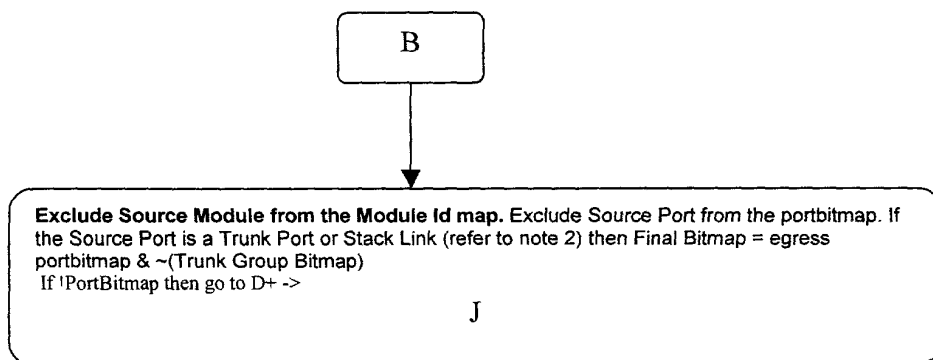


Fig. 13

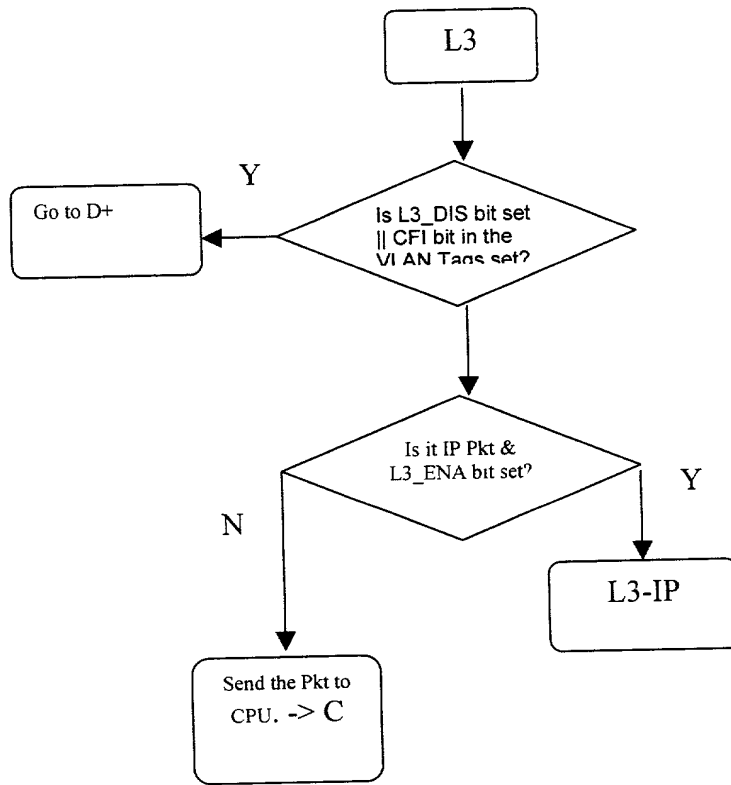


Fig. 14

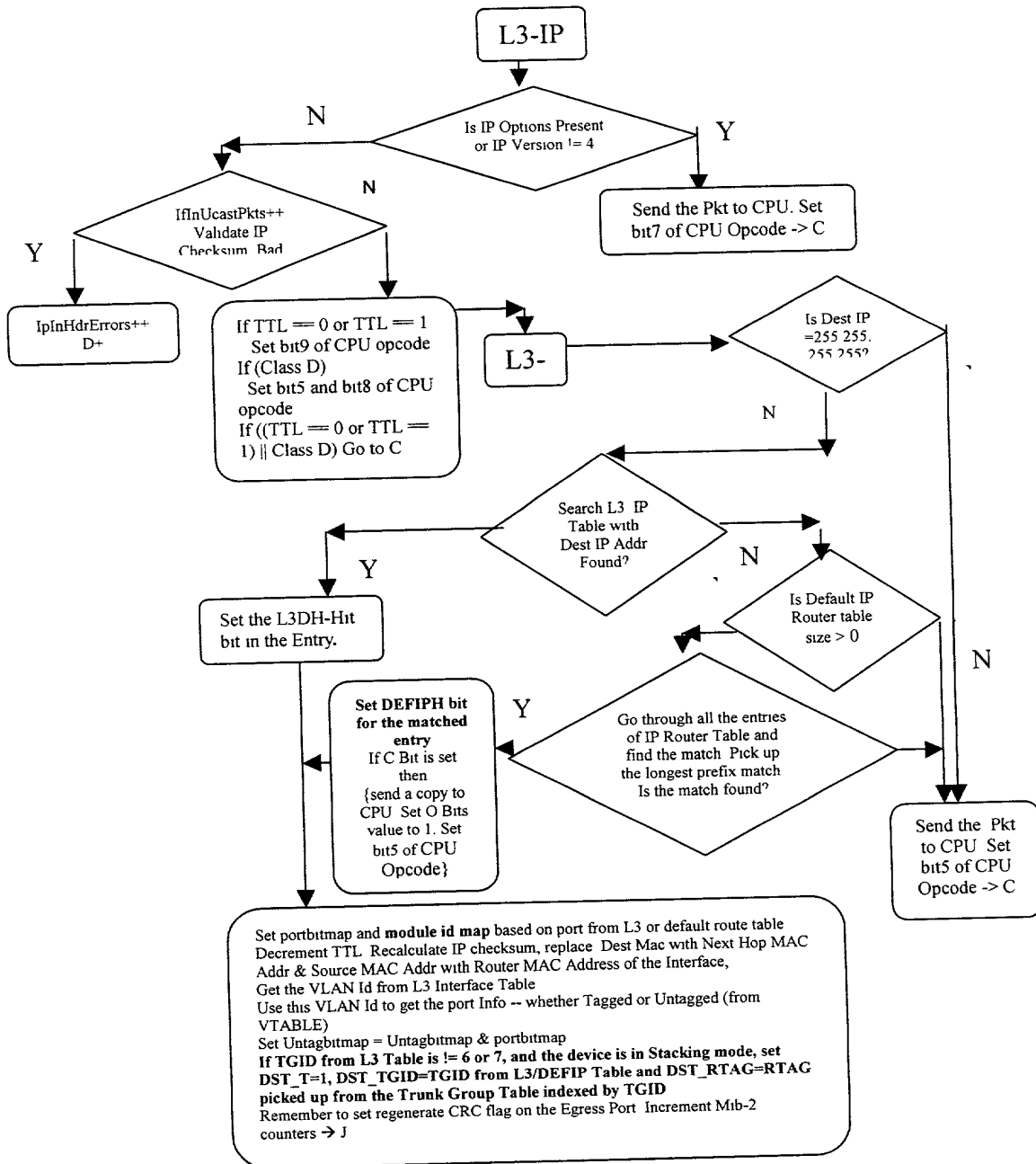


Fig. 15

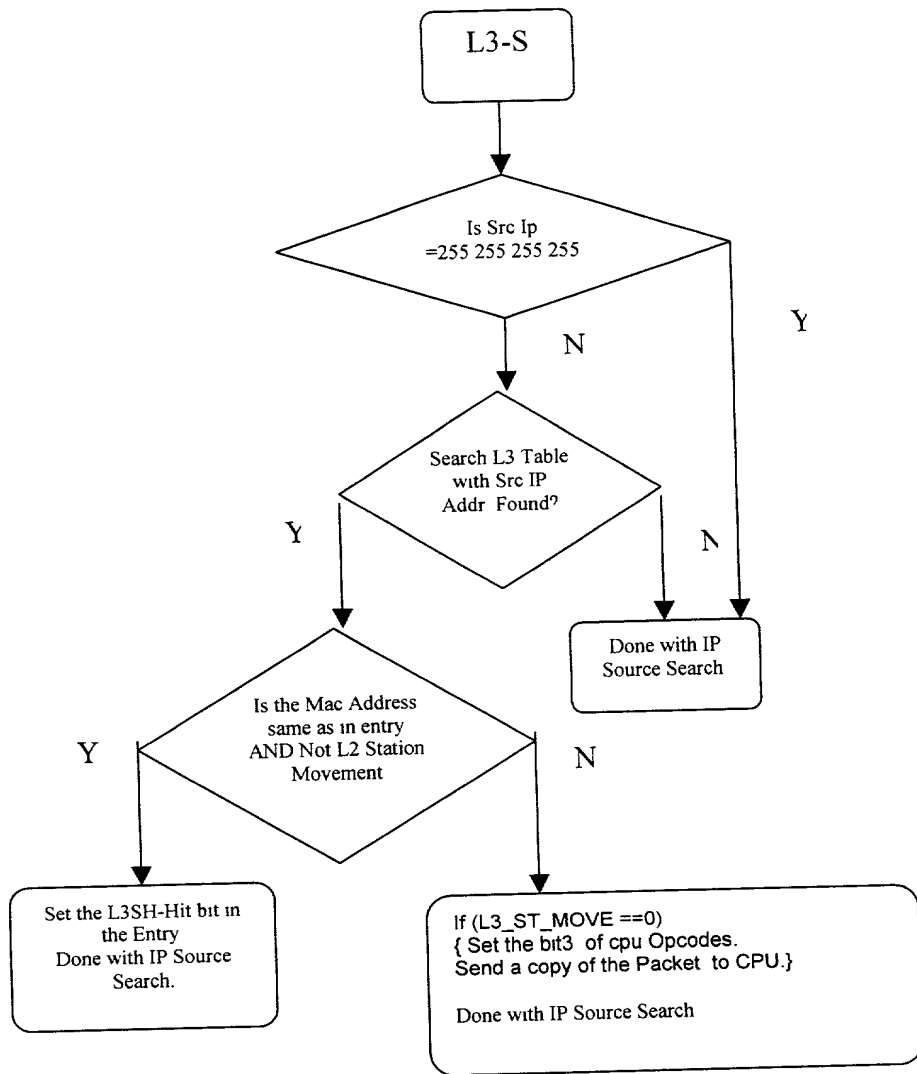


Fig. 16

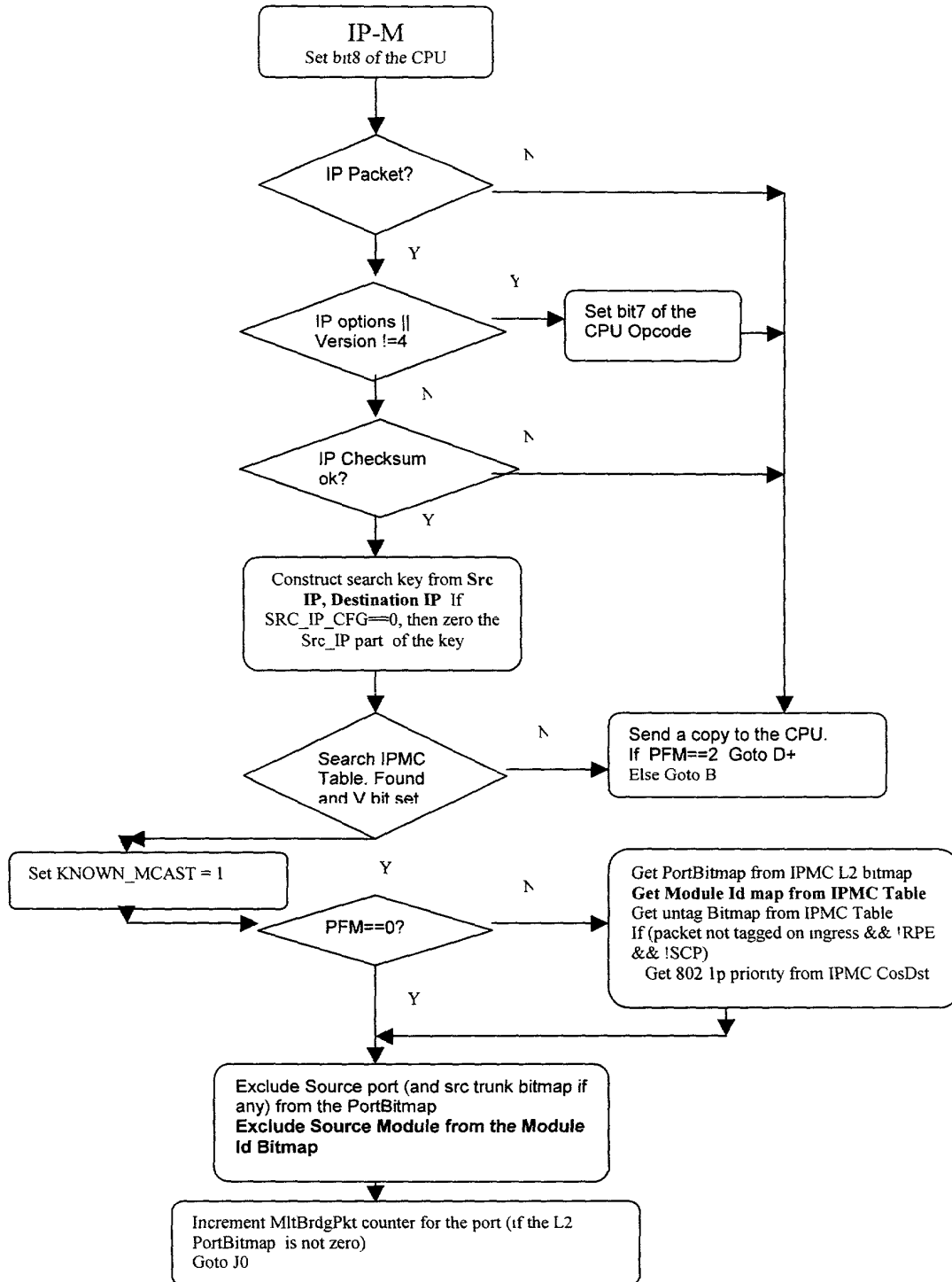


Fig. 17


```

graph TD
    J0([J0]) --> D1{Did Source search say C* or D*}
    D1 -- Y --> GotoJ1([Goto J])
    D1 -- N --> D2{L3 Mirror && Stack_link}
    D2 -- Y --> RememberMirror([Remember to Mirror])
    RememberMirror --> D3{Is L3_DIS bit set || CFI bit in the VLAN Tags set?}
    D2 -- N --> D3
    D3 -- Y --> GotoJ2([Goto J])
    D3 -- N --> ProcessBlock1
    subgraph ProcessBlock1 [ ]
        direction TB
        P1_1[If (!Class D), set CPU Opcodes bit5]
        P1_2[If TTL <= threshold from the IPMC table]
        P1_3[Then set CPU opcodes bit9]
        P1_4[If (!Class D || TTL <= threshold) {]
        P1_5[Send a copy to CPU. Go to J}]
        P1_4 --- P1_5
    end
    ProcessBlock1 --> L3S([L3-S])
    L3S --> ProcessBlock2
    subgraph ProcessBlock2 [ ]
        direction TB
        P2_1[Port_Match = TRUE]
        P2_2[If (SP_Check) {]
        P2_3[Port_Match=FALSE]
        P2_4[If (VID of the packet != VID from the Table) Port_Match = FALSE]
        P2_5[else {]
        P2_6[If (T bit of Source port == TS) {]
        P2_7[If (T bit of the Source port) {]
        P2_8[If (TGID of the Source port == port_TGID from IPMC table) Port_Match=TRUE}]
        P2_9[Else if (Source port == port_TGID from IPMC Table) Port_Match =TRUE}]
        P2_10[}]
        P2_11[}]
        P2_12[}]
    end
    ProcessBlock2 --> D4{Port_Match?}
    D4 -- N --> ProcessBlock3
    subgraph ProcessBlock3 [ ]
        direction TB
        P3_1[If (CD) send a copy to CPU]
        P3_2[Goto J]
    end
    ProcessBlock3 --> D4
    D4 -- Y --> ProcessBlock4
    subgraph ProcessBlock4 [ ]
        direction TB
        P4_1[Get L3 PortBitmap from the IPMC Table]
        P4_2[Increment MltRoutPkt counter for the port (if the corresponding L3 PortBitmap is not zero)]
        P4_3[PortBitmap = PortBitmap | L3 PortBitmap]
        P4_4[Goto J]
    end

```

Fig. 18

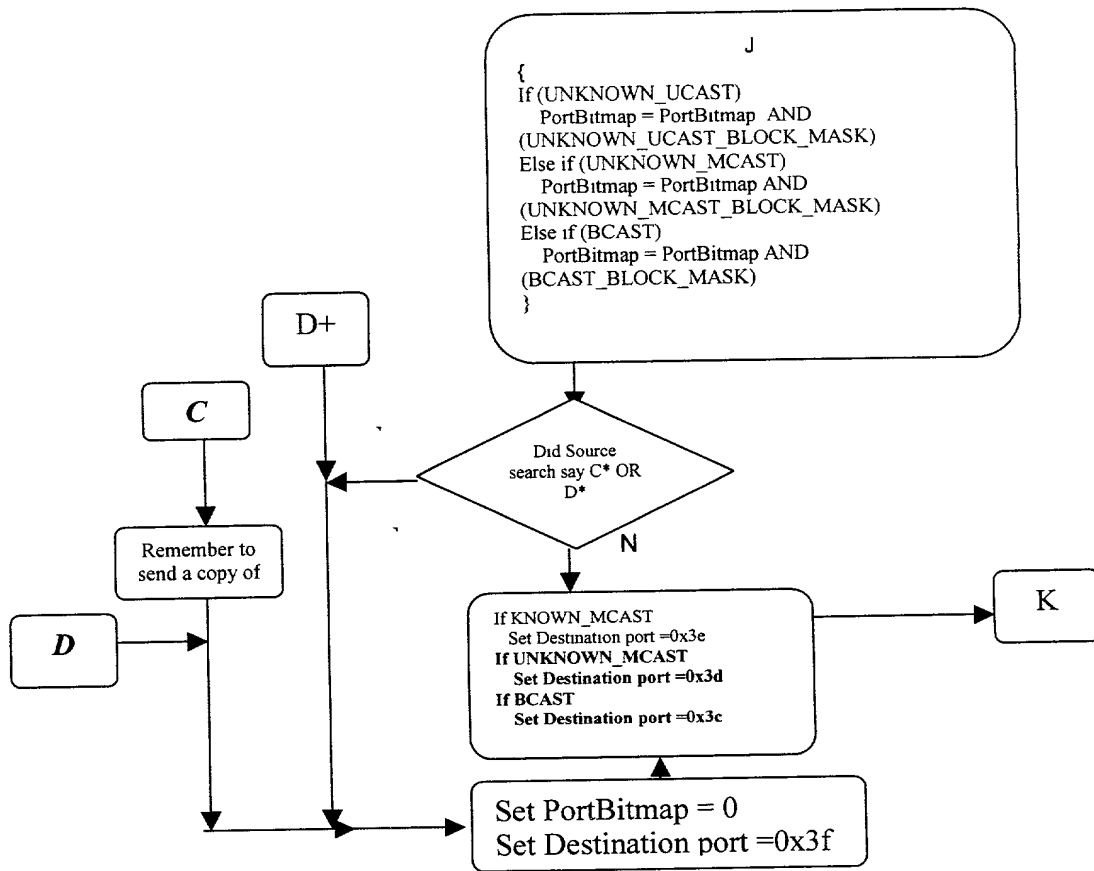


Fig. 19

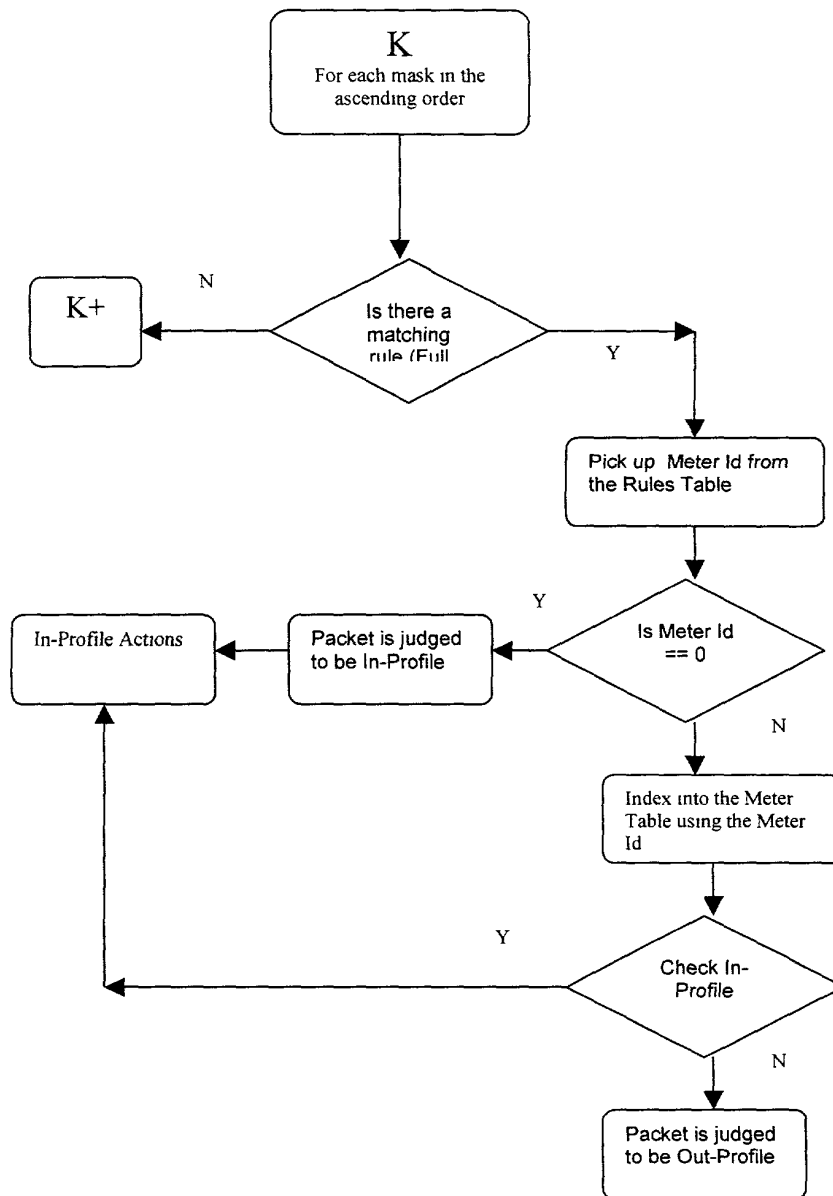


Fig. 20

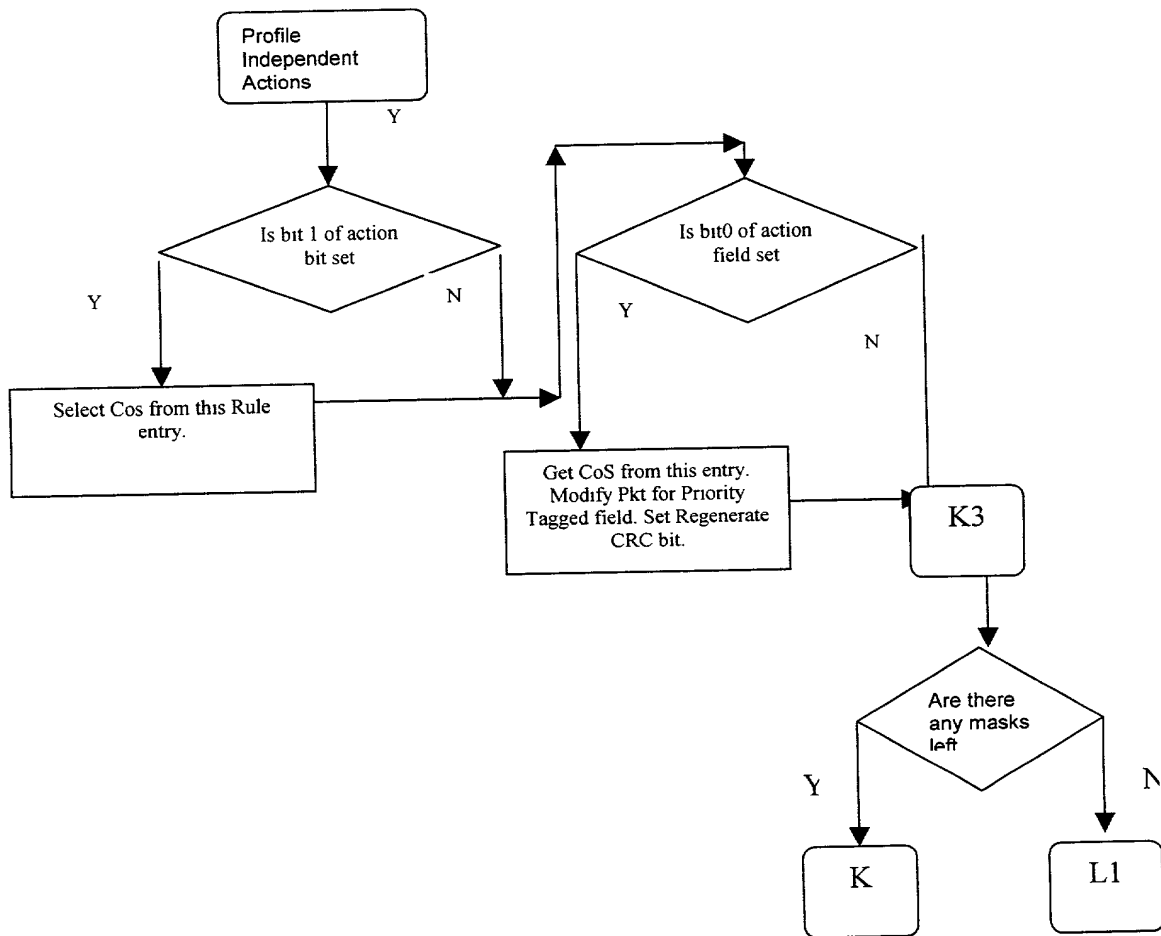


Fig. 21

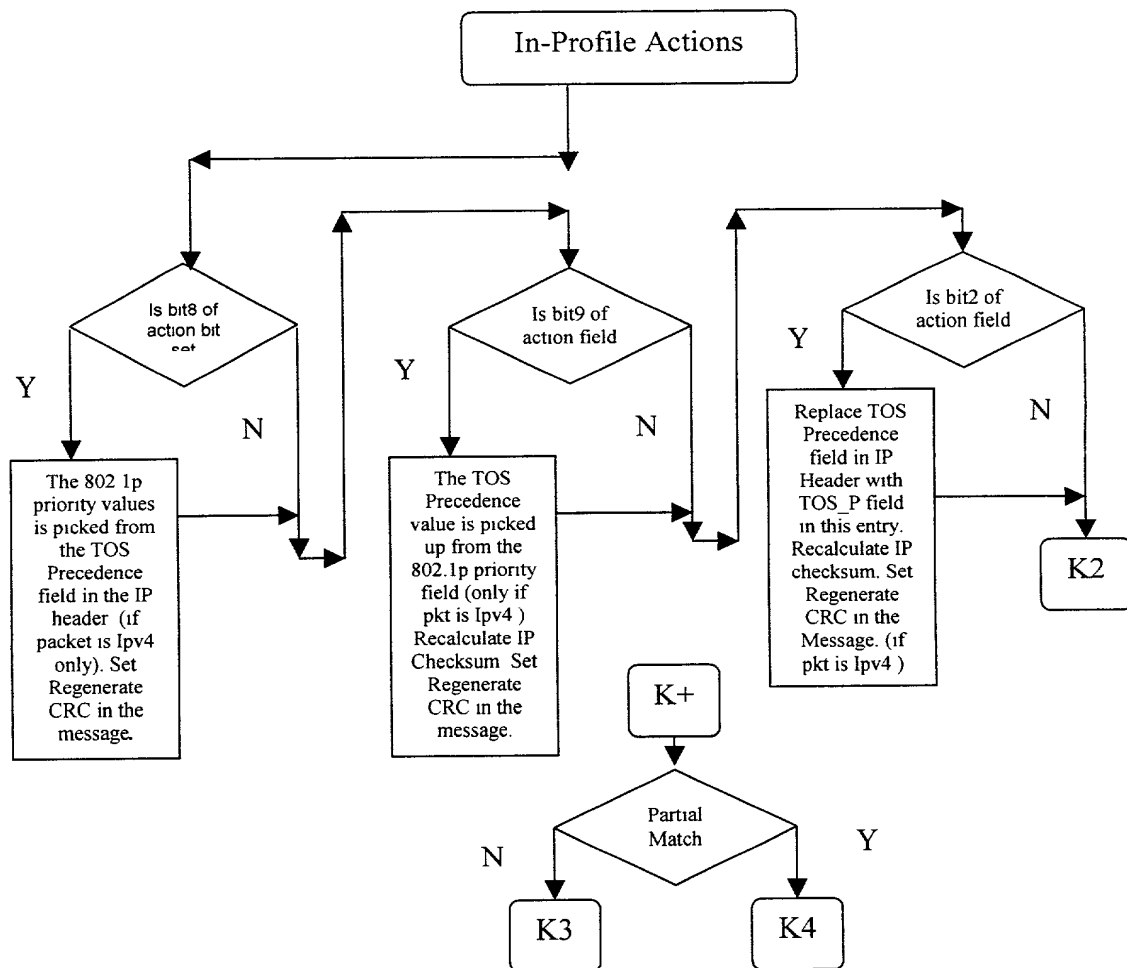


Fig. 22

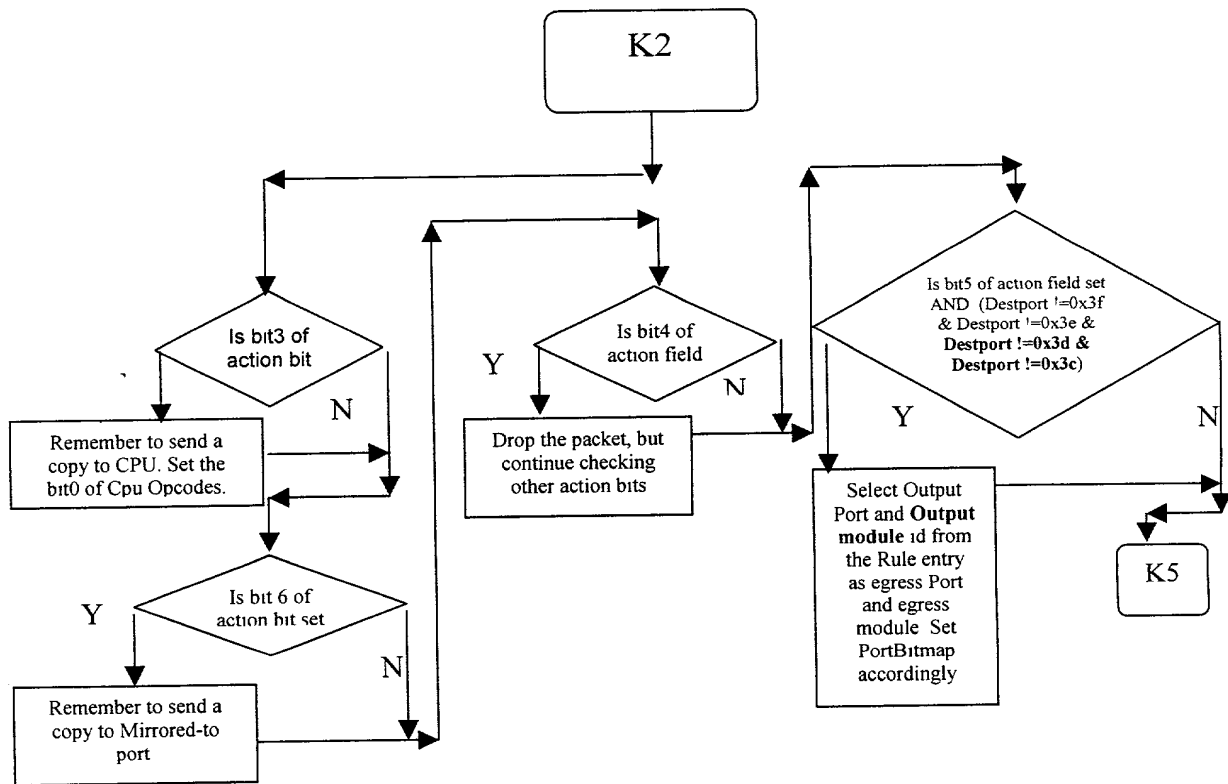


Fig. 23

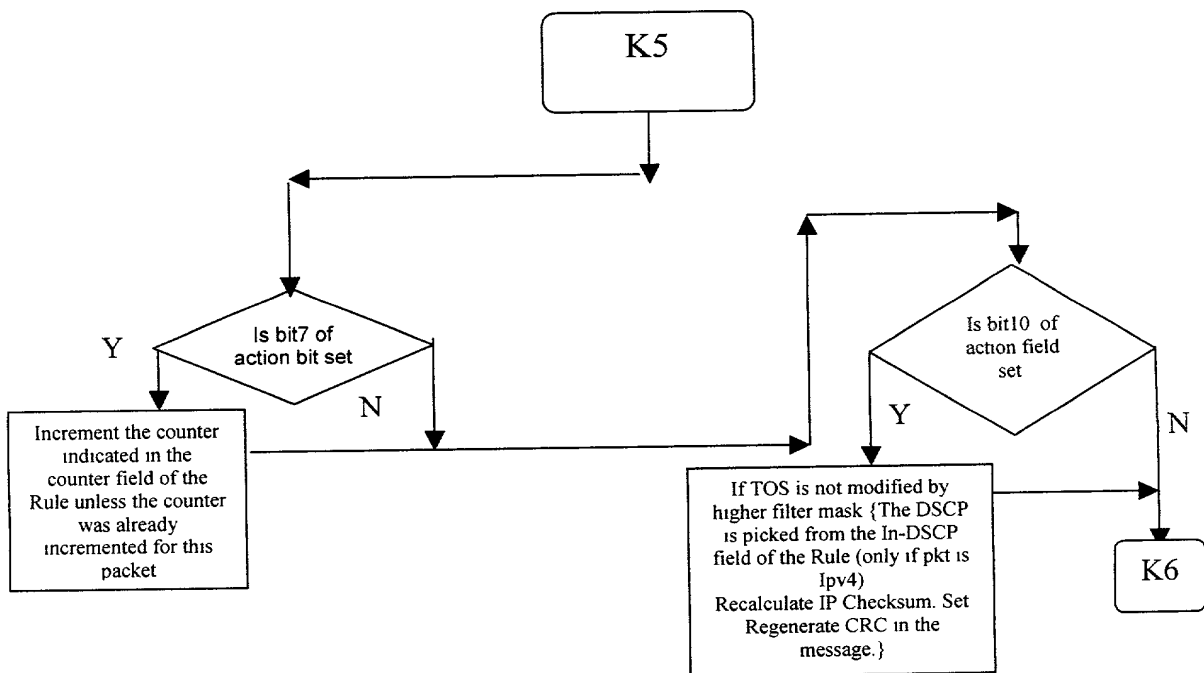


Fig. 24

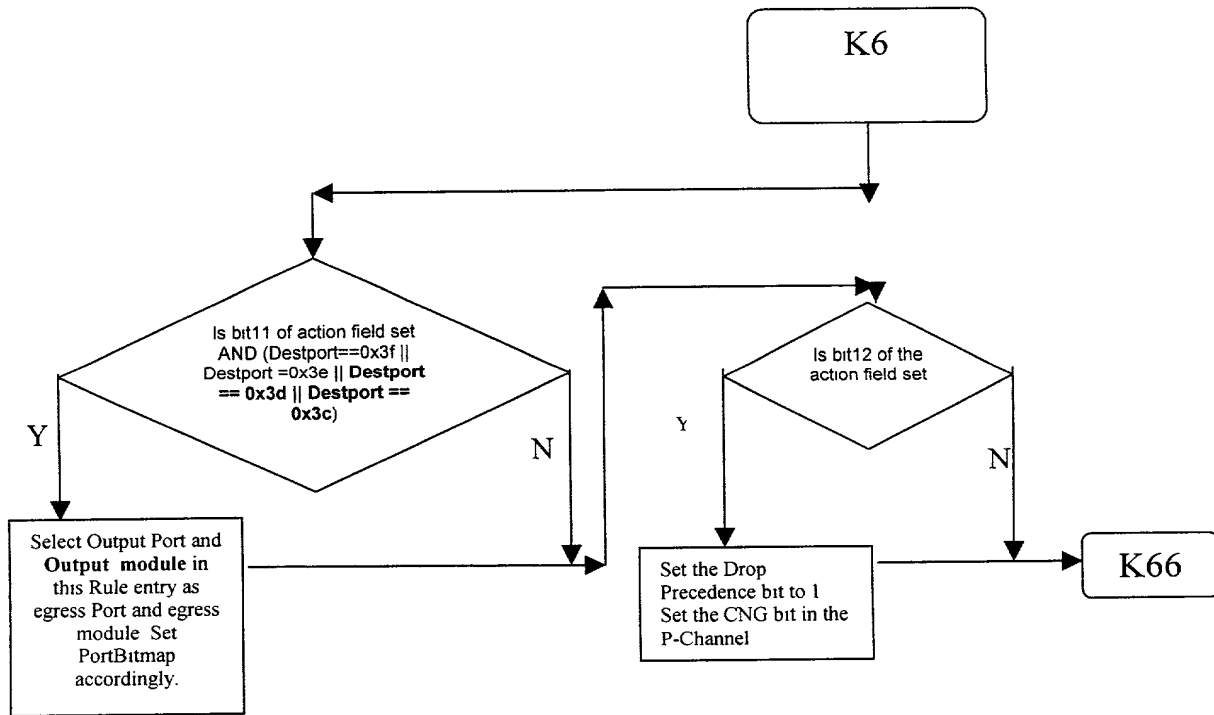


Fig. 25

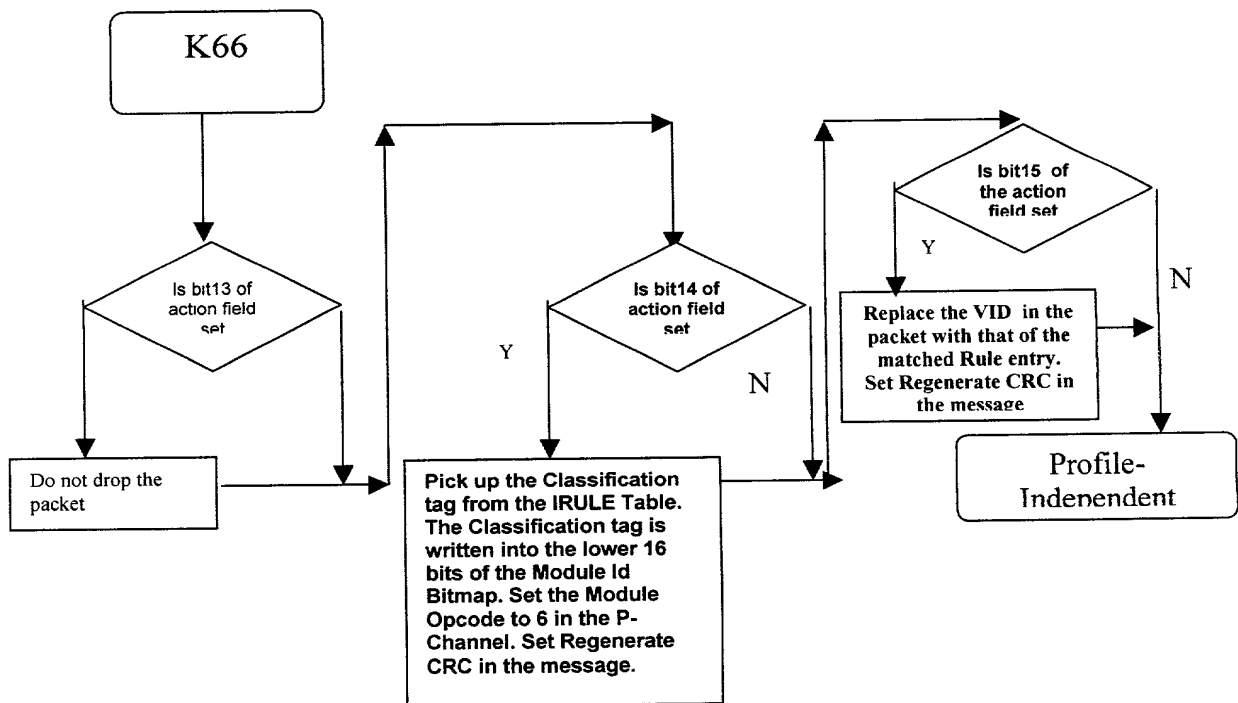


Fig. 26

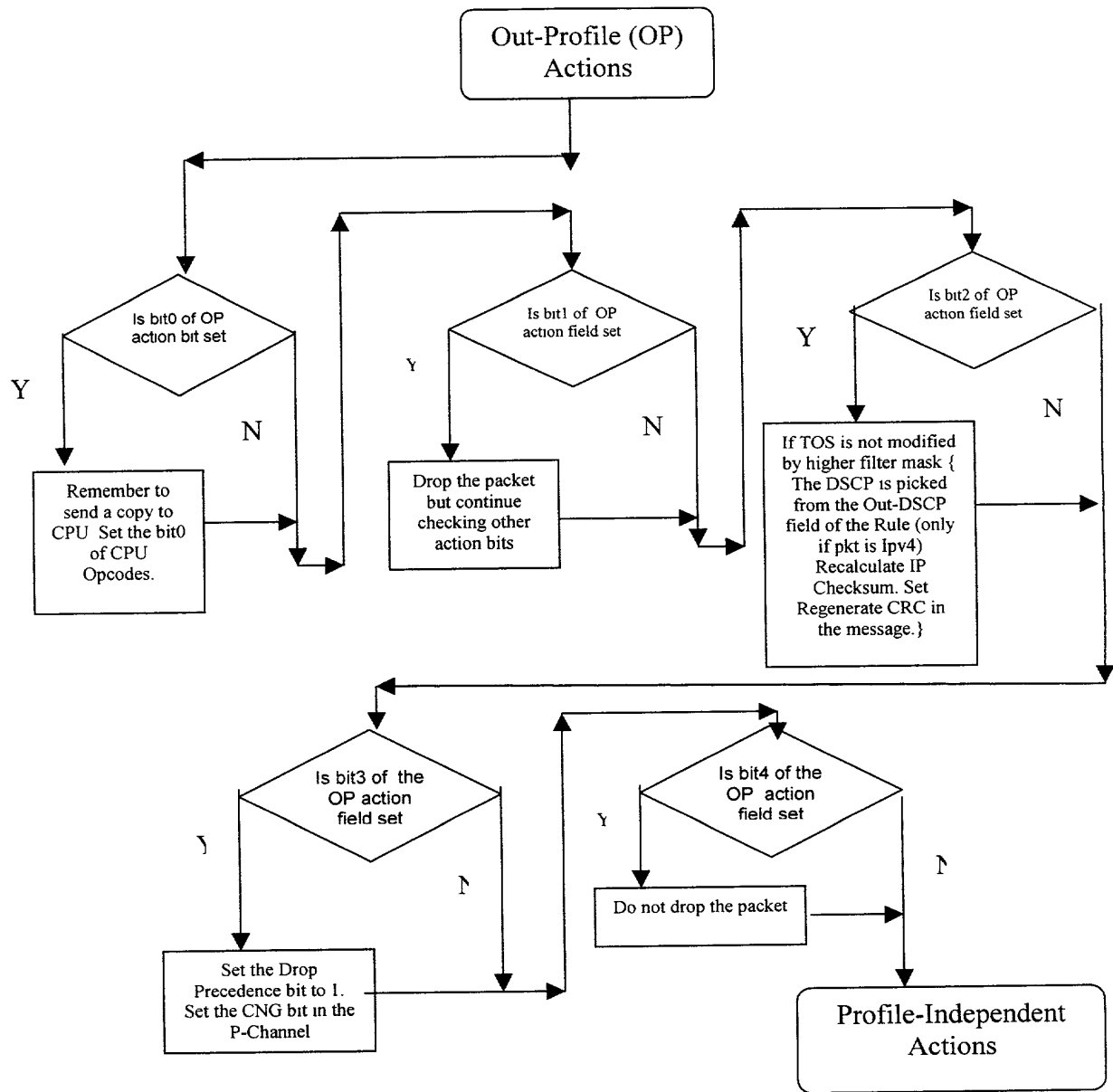


Fig. 27

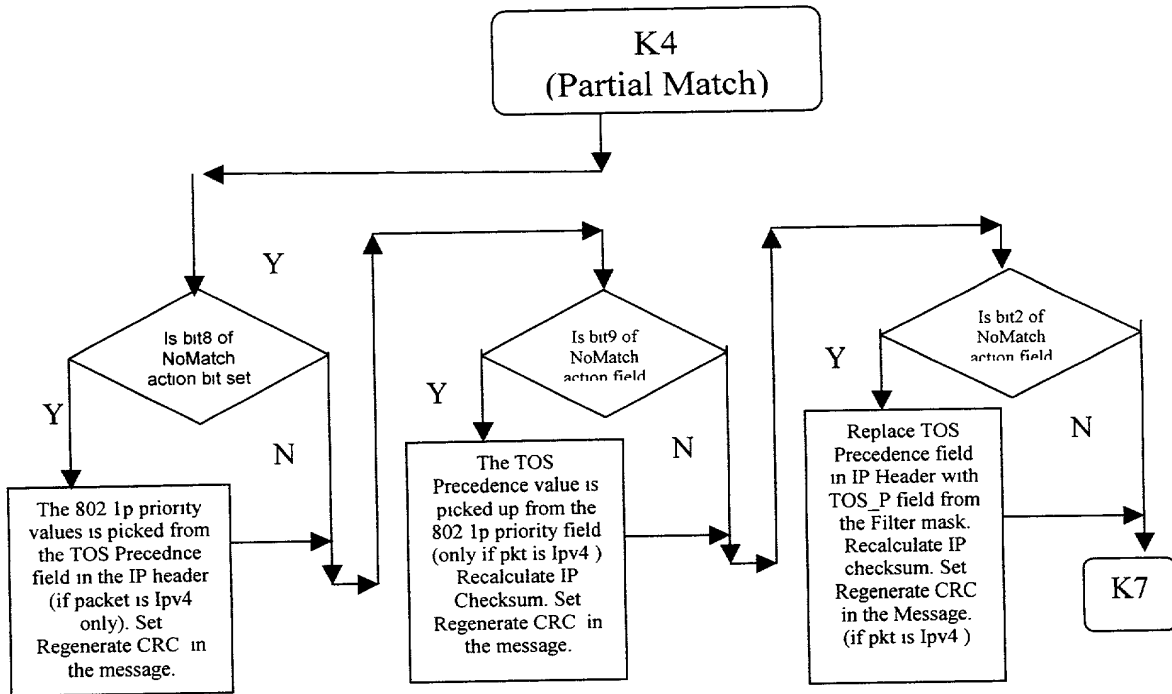


Fig. 28

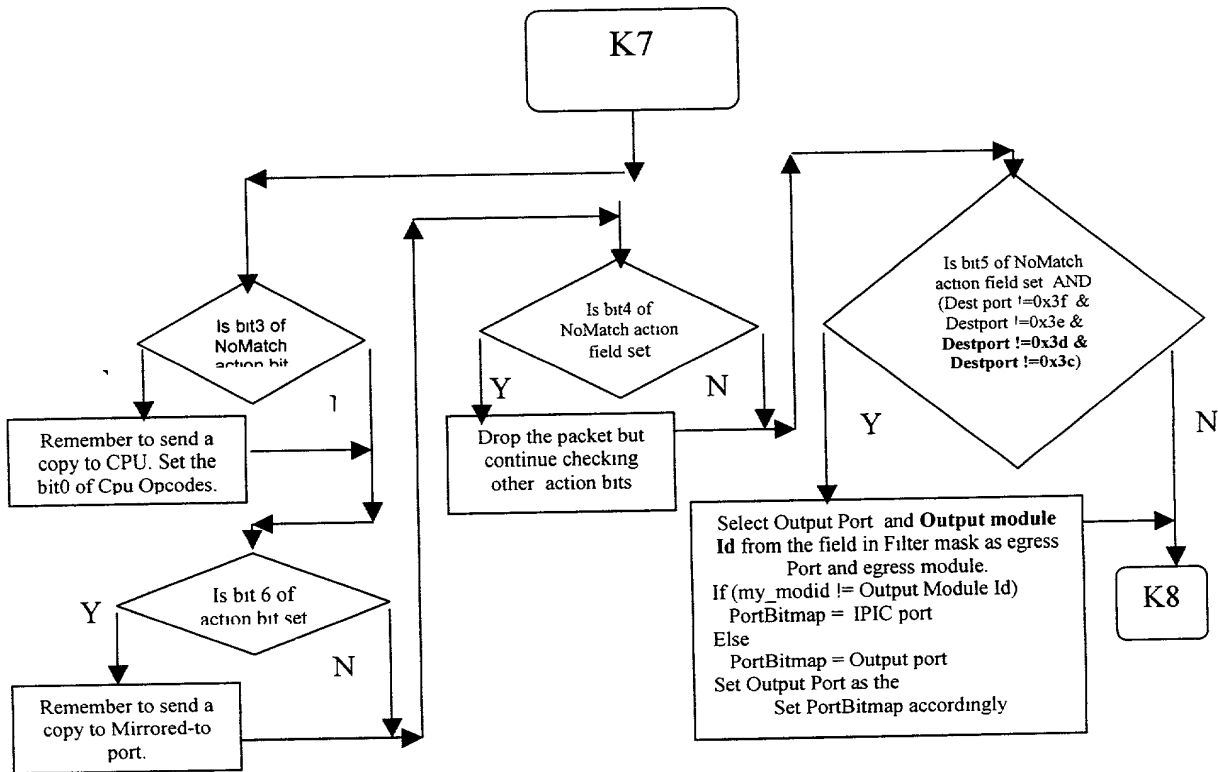


Fig. 29

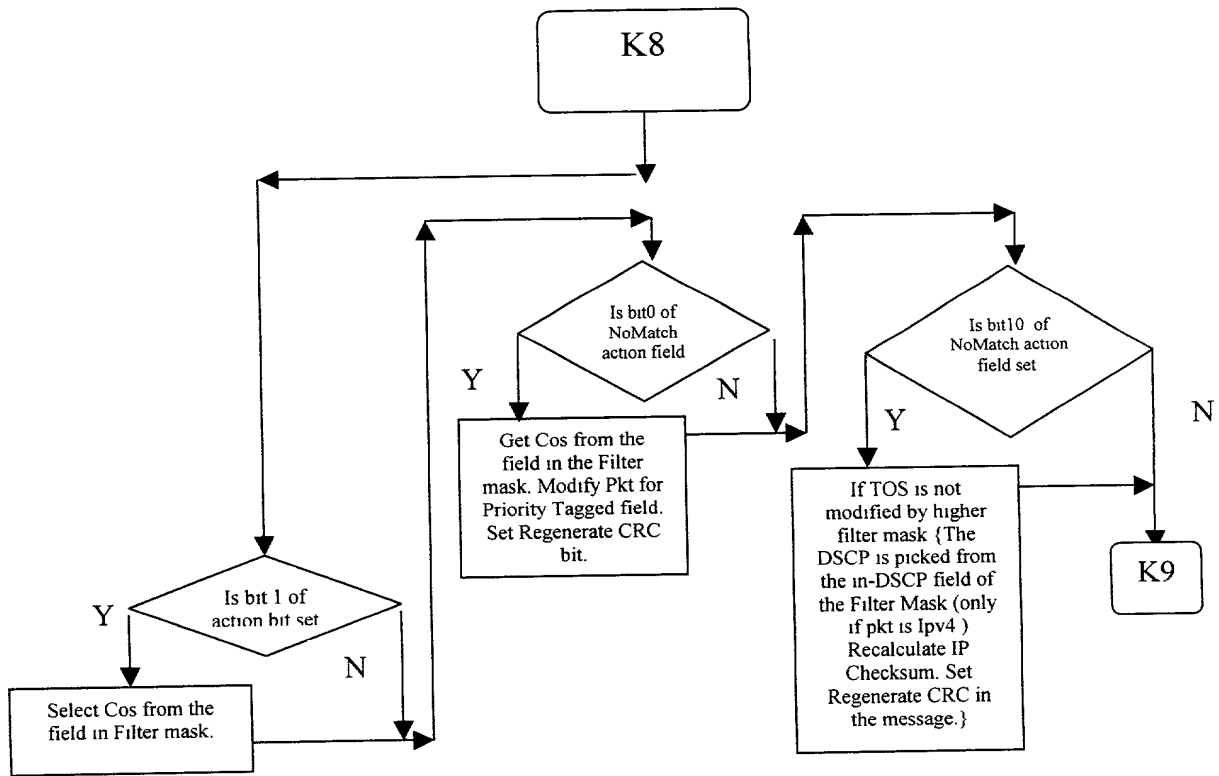


Fig. 30

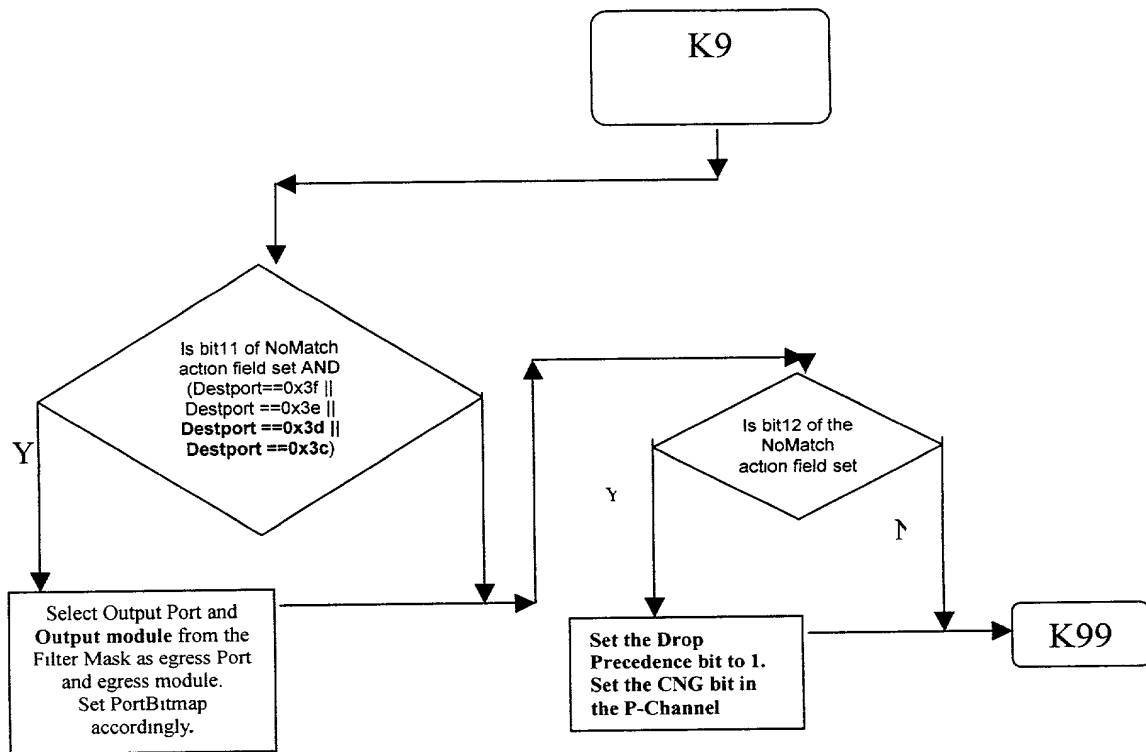


Fig. 31

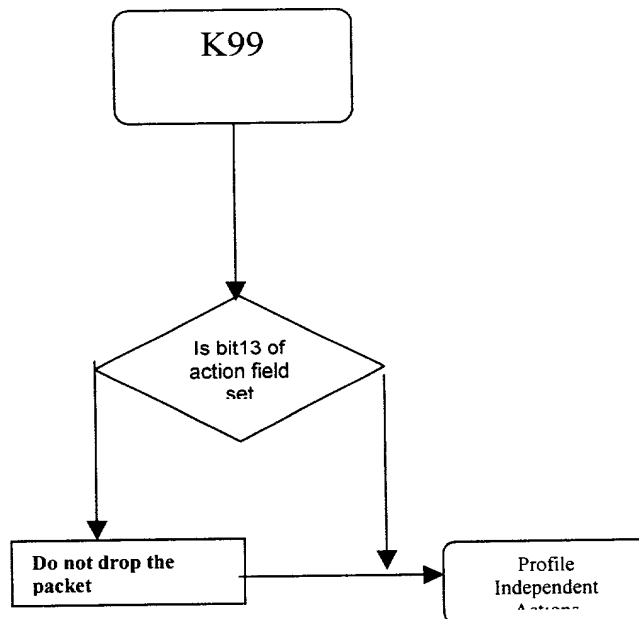


Fig. 32

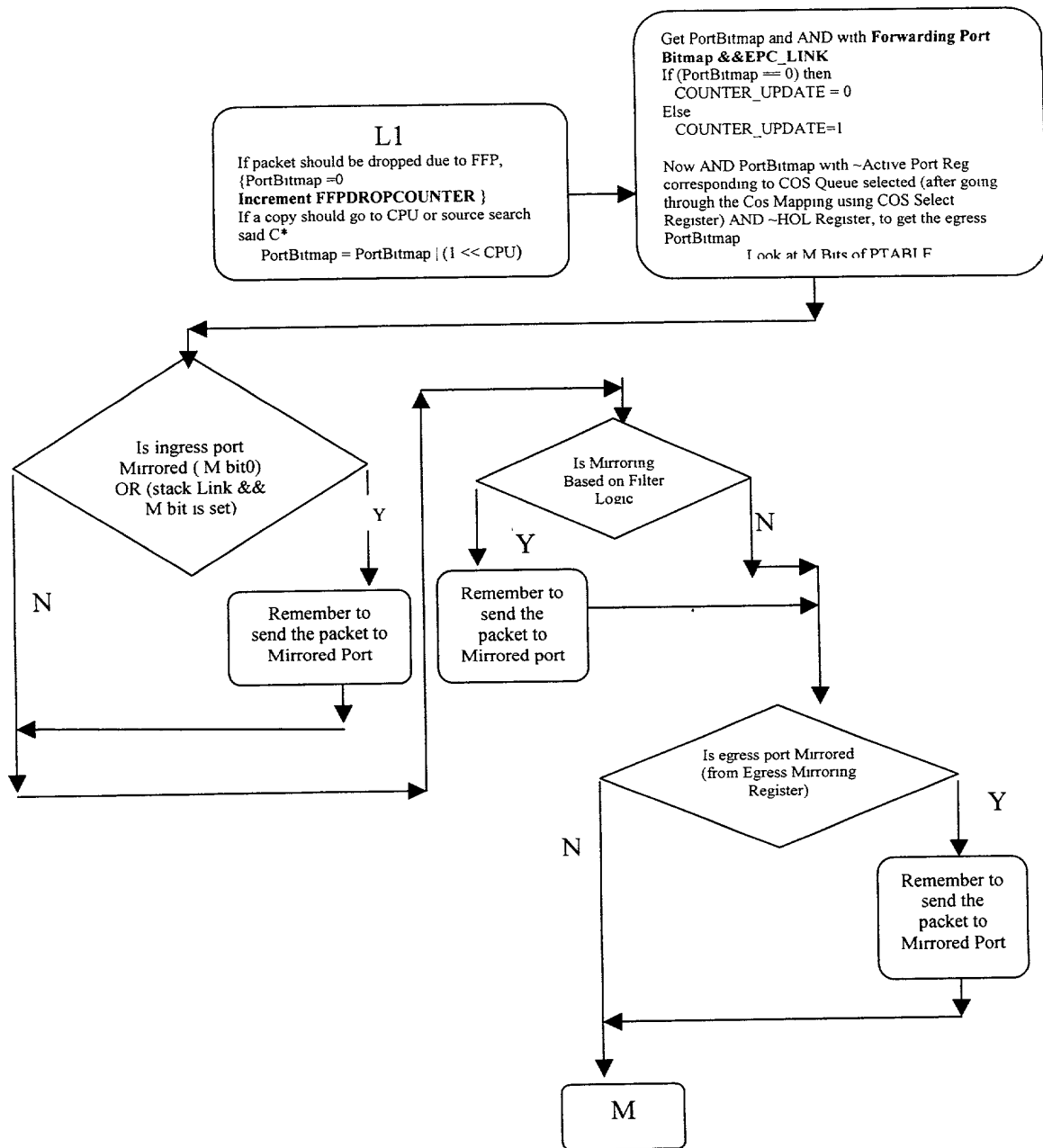


Fig. 33

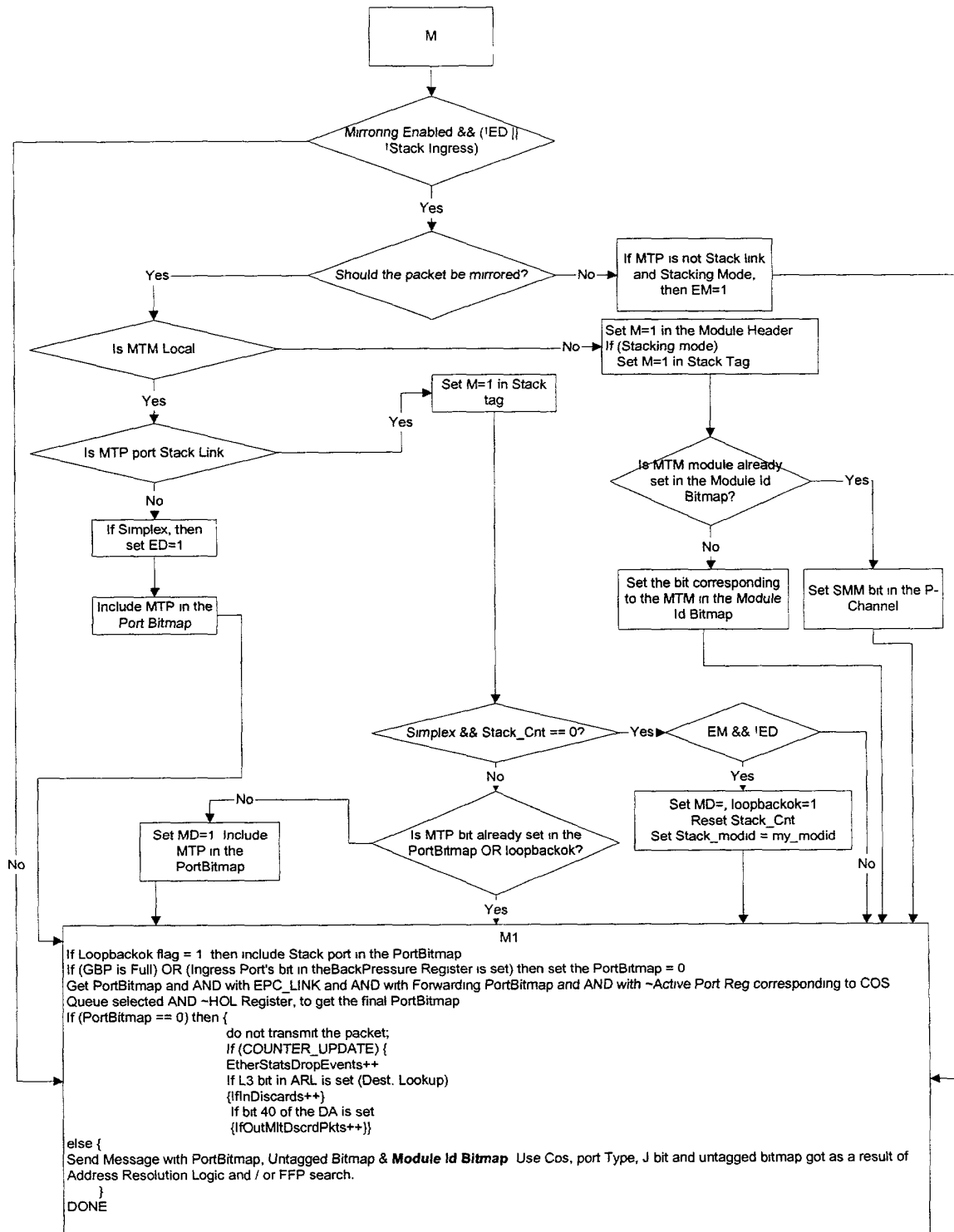


Fig. 34

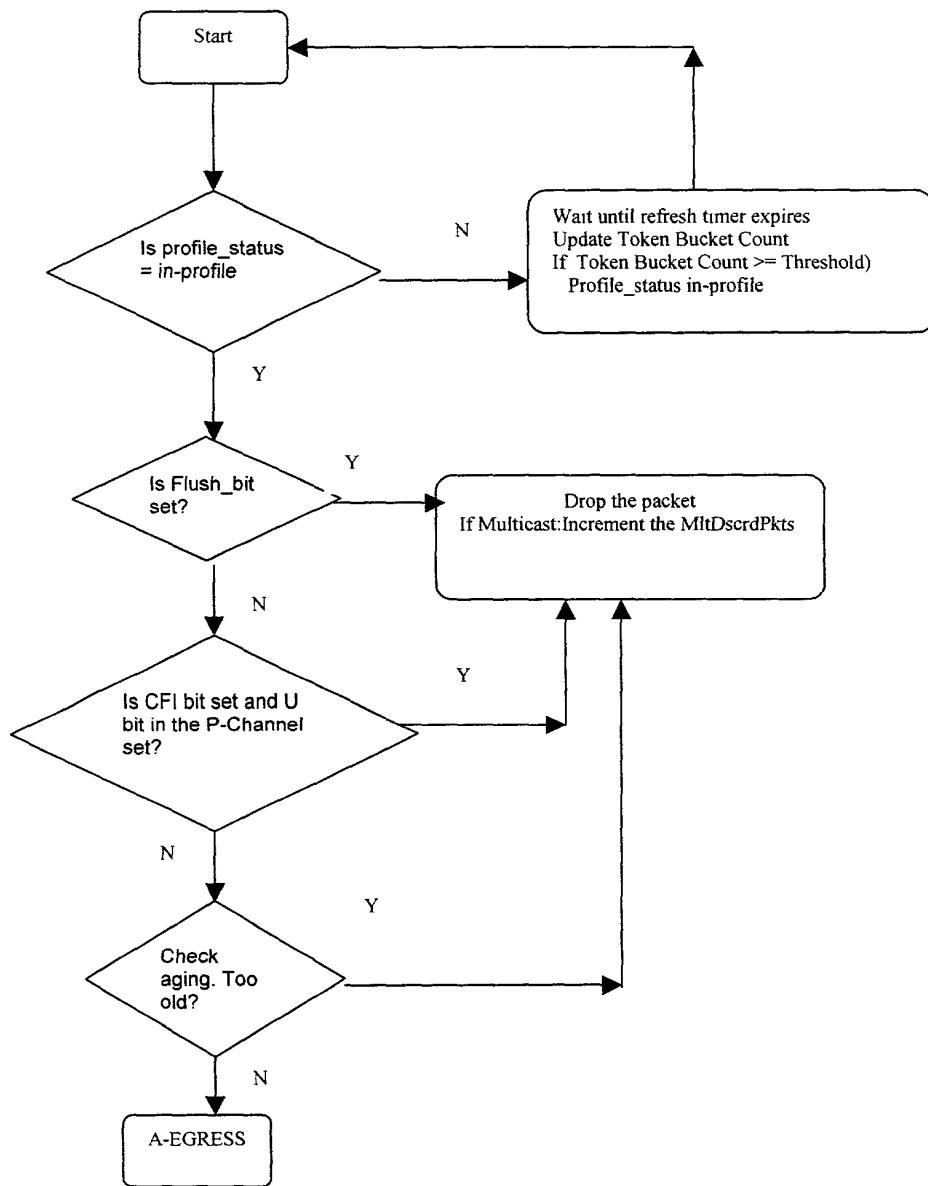


Fig. 35

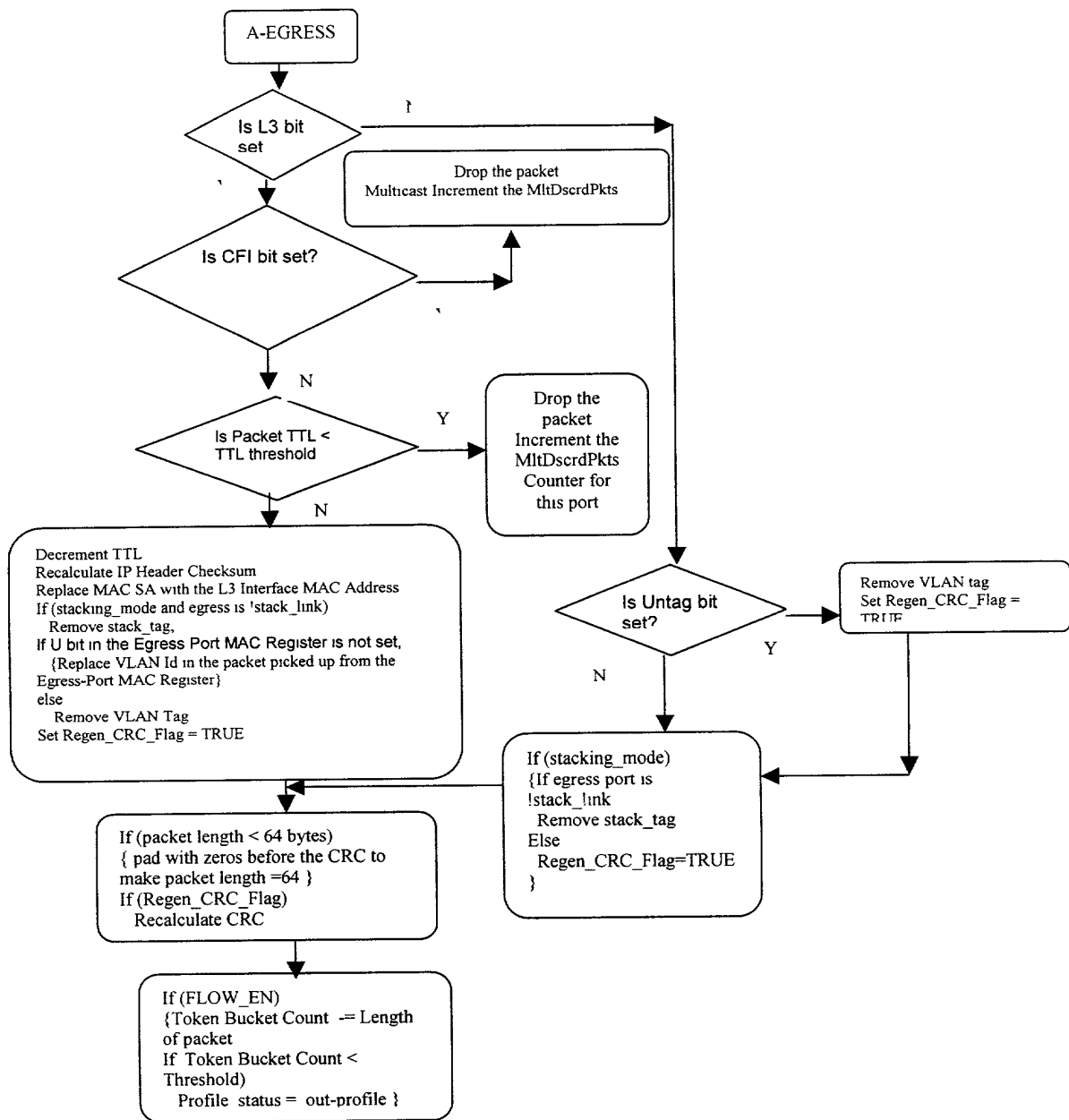


Fig. 36

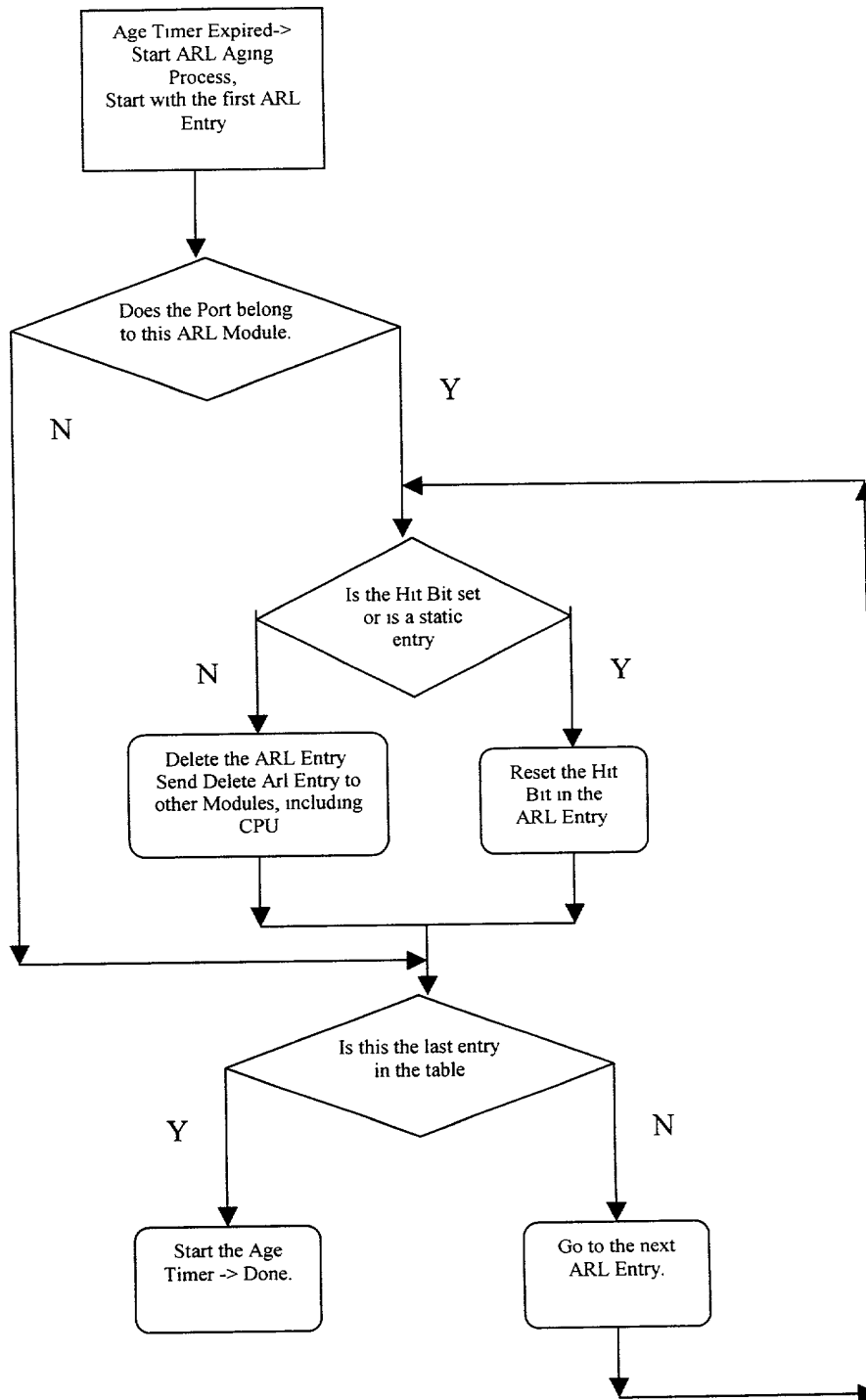


Fig. 37

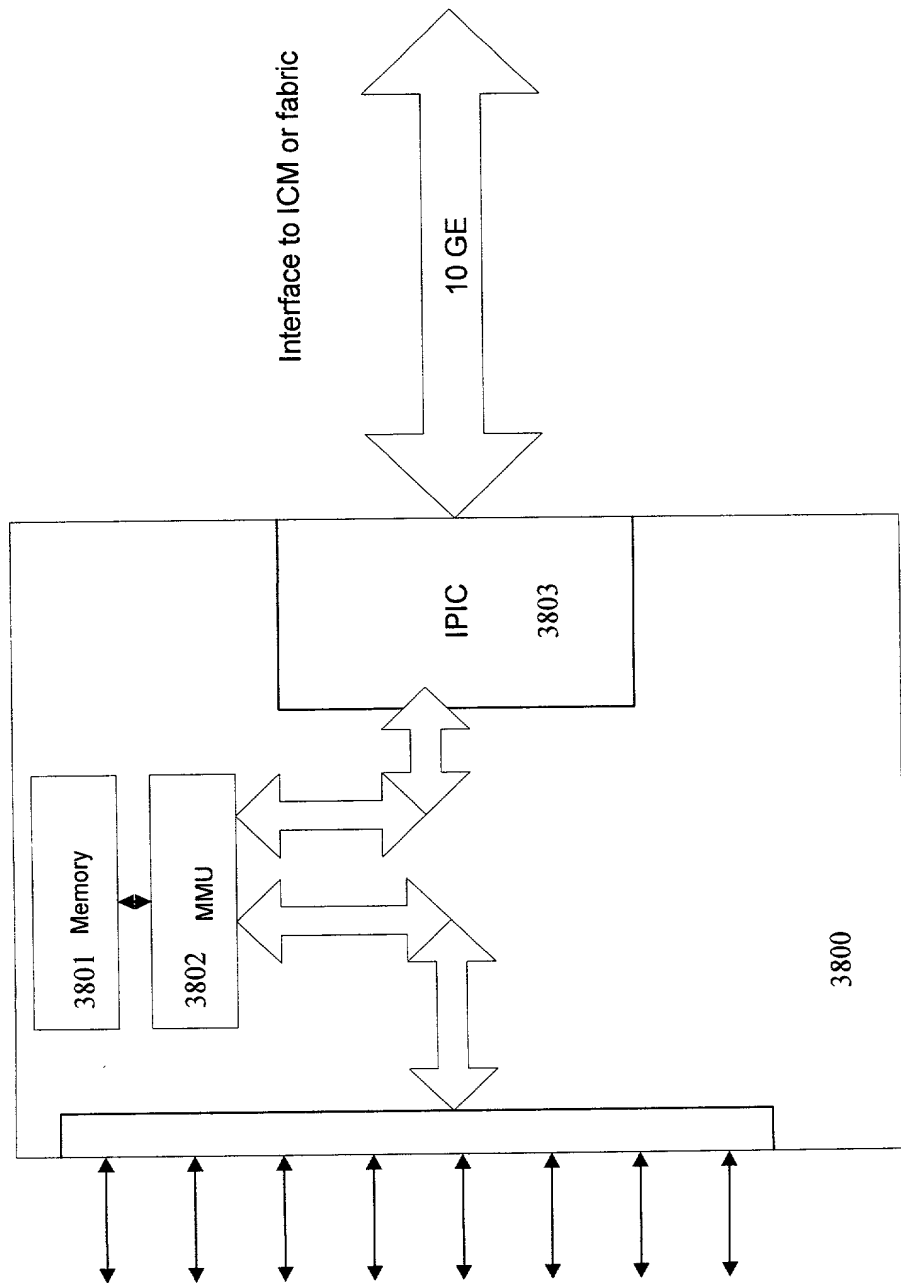


Fig. 38

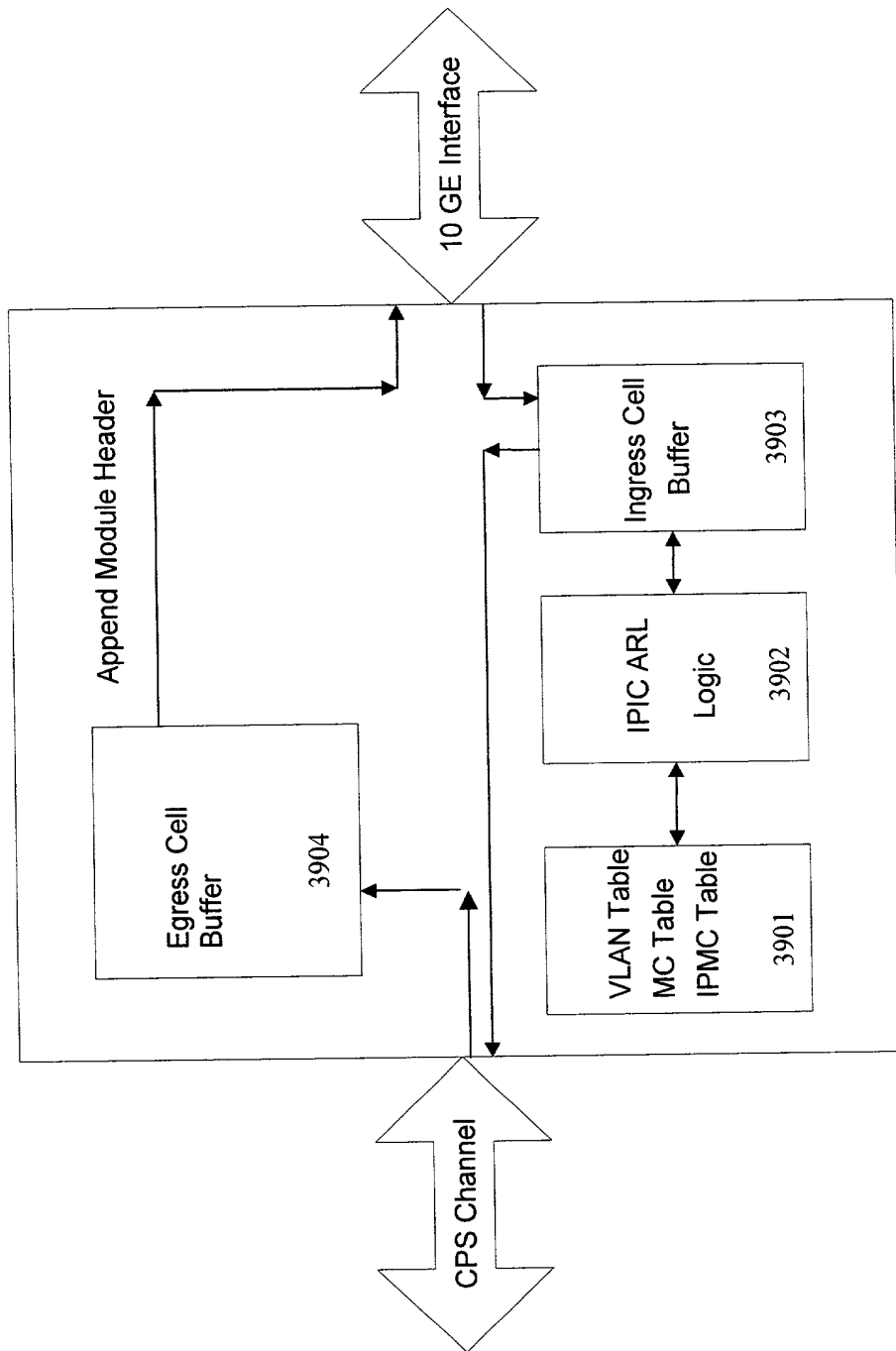


Fig. 39

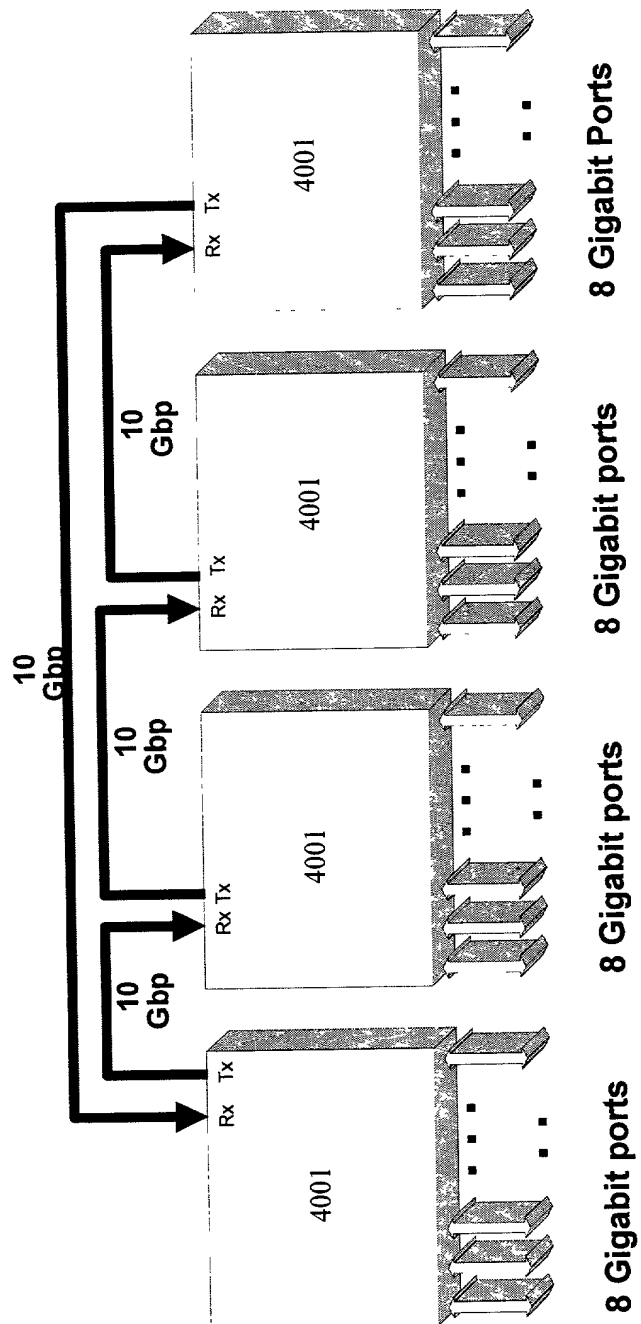


Fig. 40

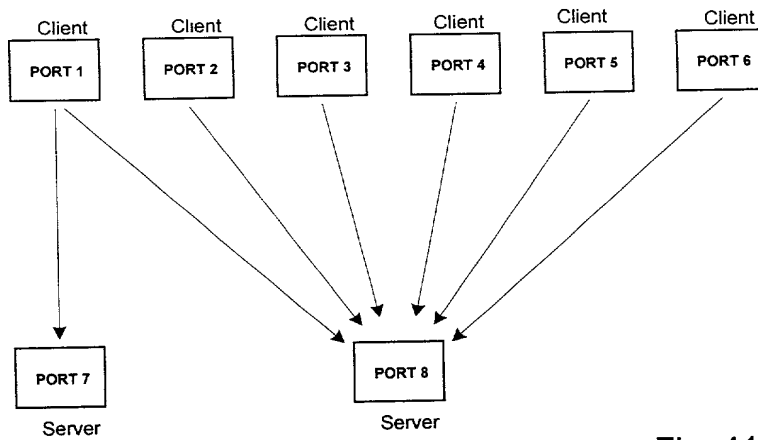


Fig. 41

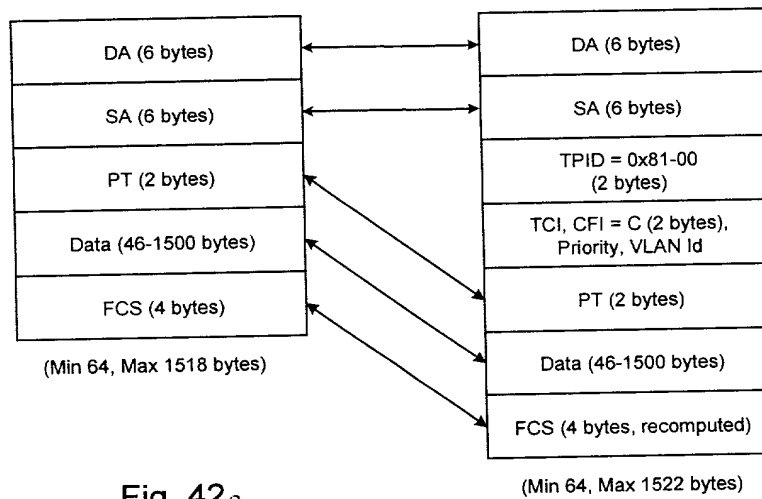


Fig. 42 a

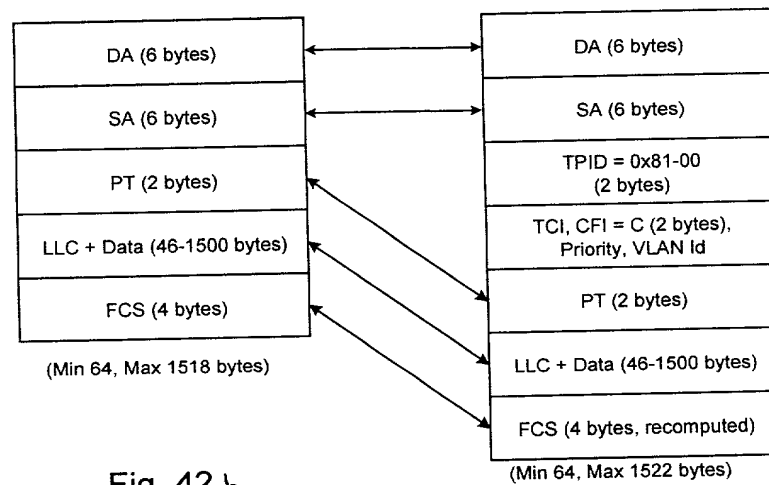


Fig. 42 b

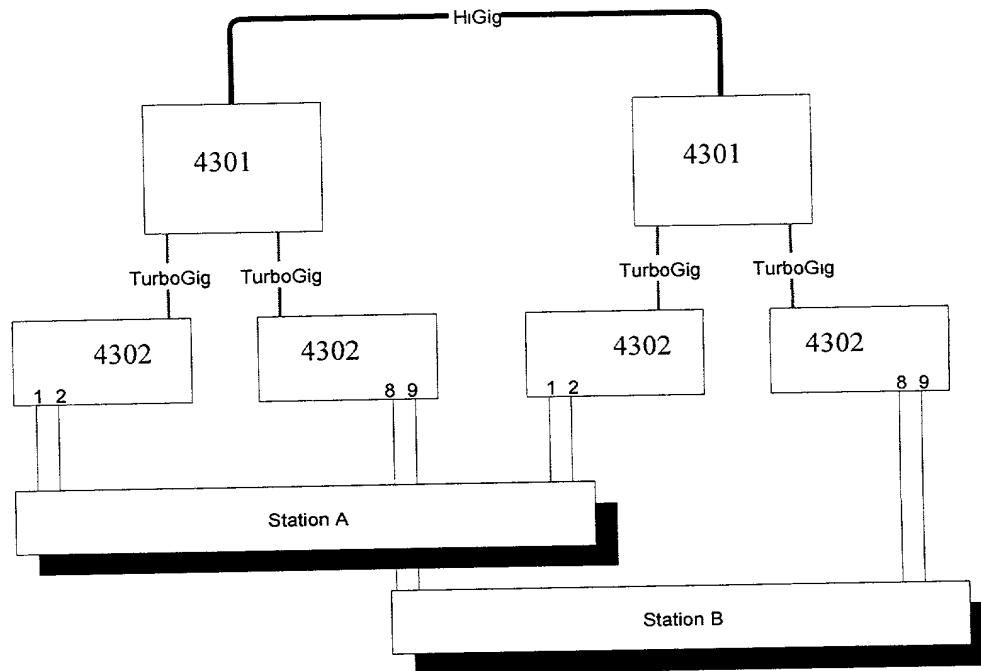


Fig. 43

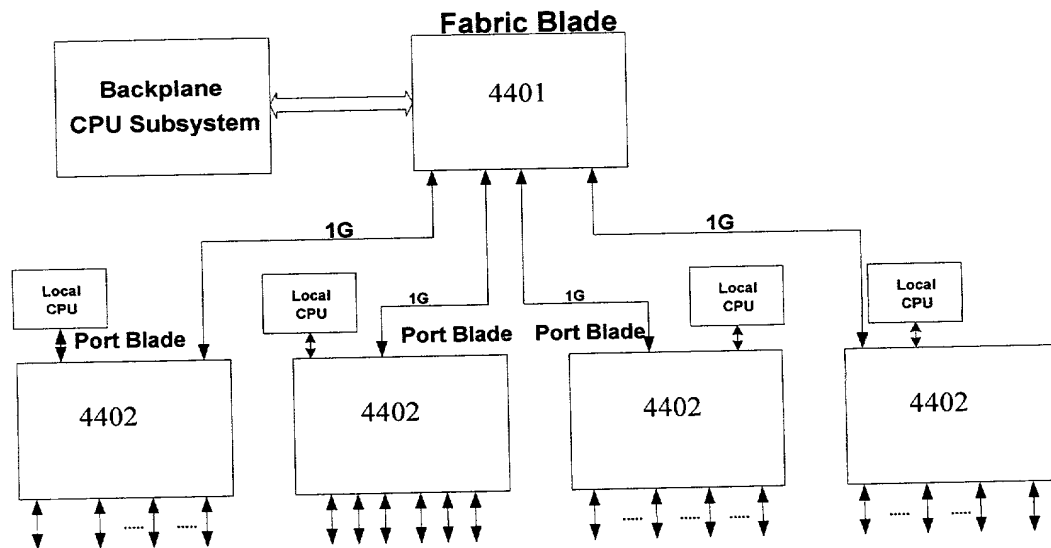


Fig. 44

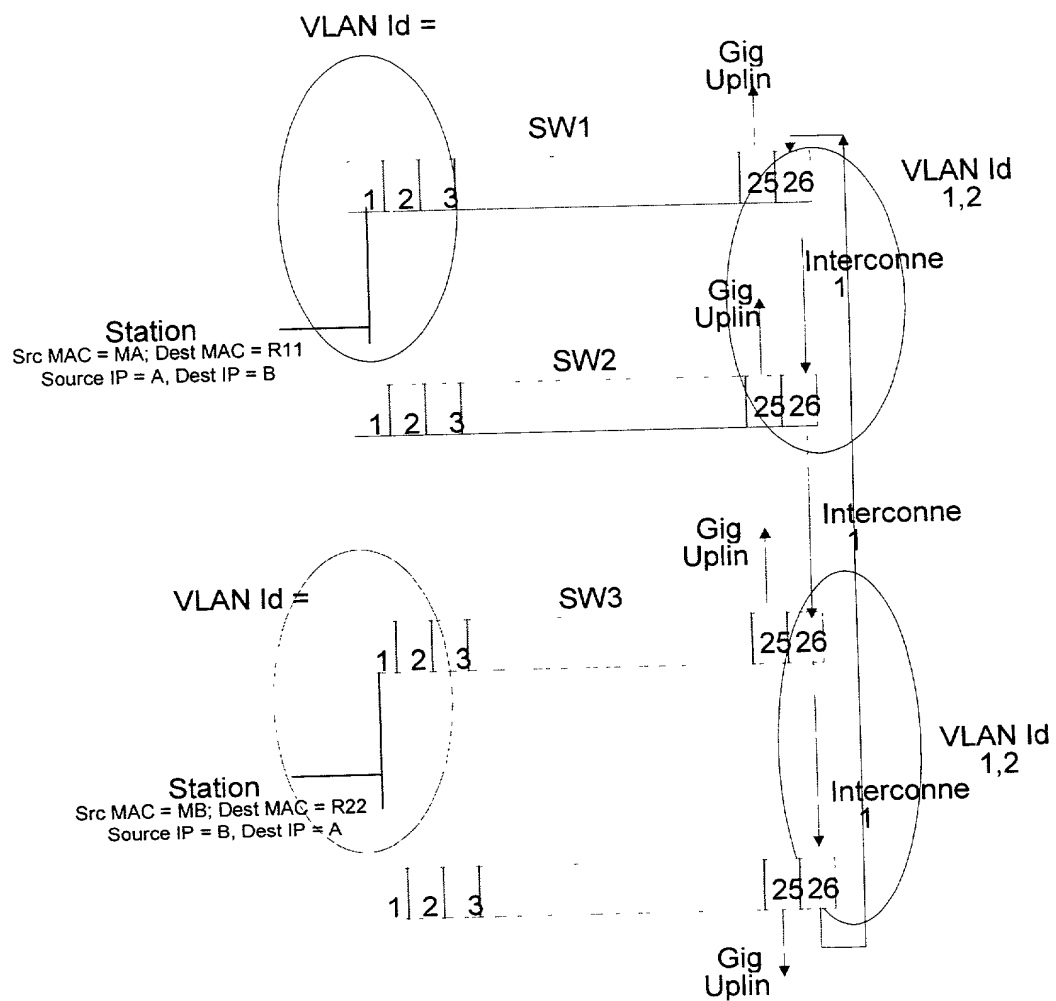


Fig. 45

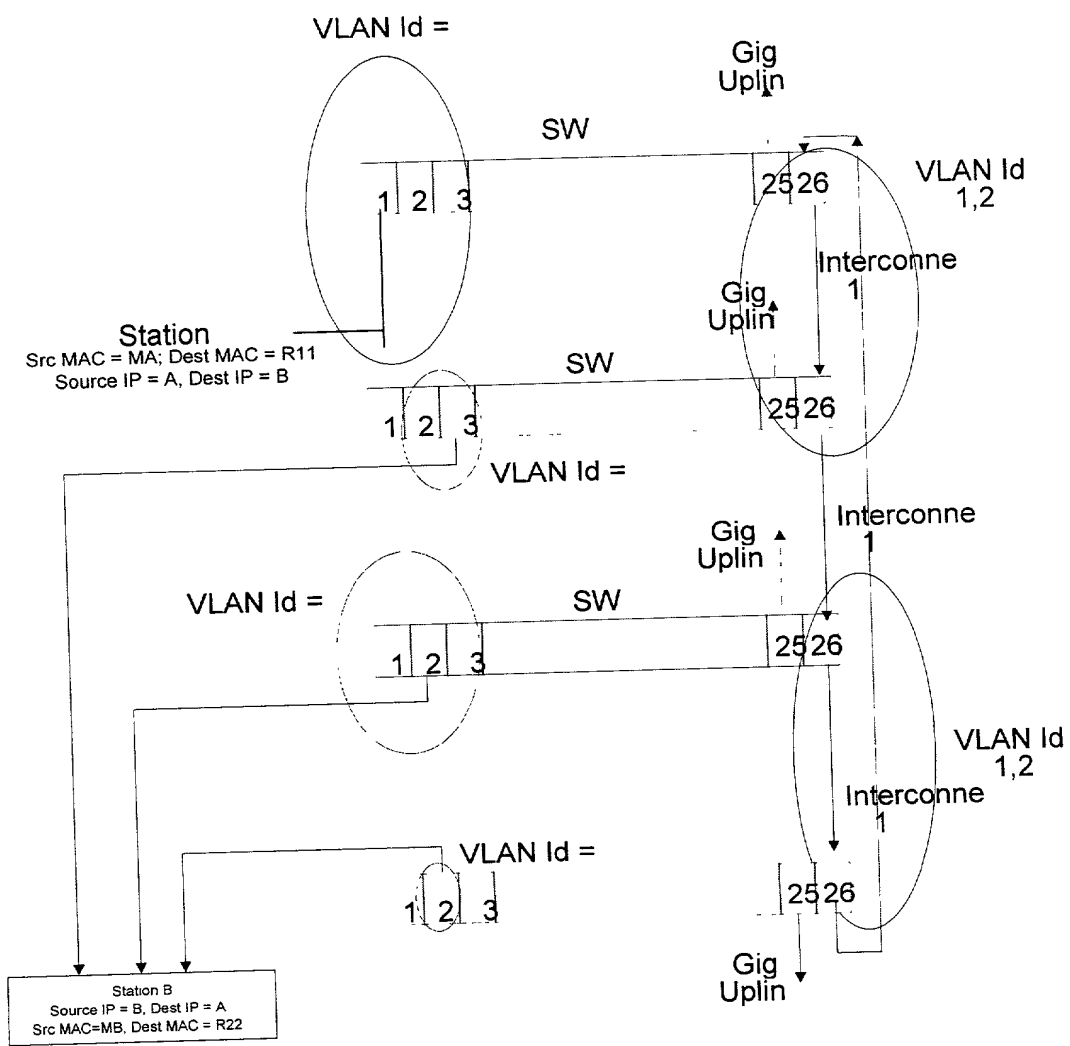


Fig. 46